The Curriculum Book

BACHELOR OF TECHNOLOGY

in

COMPUTER SCIENCE AND ENGINEERING

FOUR YEAR PROGRAMME

Choice Based Credit System

w. e. f. July 2019

(70:30)

Approved by Faculty (E&T) for 2018 onwards batch
18/7/19

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

GURU JAMBHESHWAR UNIVERSITY OF

SCIENCE & TECHNOLOGY

HISAR-125001, HARYANA

GURU JAMBHESHWAR UNIVERSITY OF SCIENCE & TECHNOLOGY, HISAR
(Established by State Legislature Act 17 of 1995)
'A' Grade, NAAC Accredited State Govt. University

Acad./AC-III/Fac-1 Vol.3/2019/ 4199
Dated: 20/8/19

To

The Controller of Examinations
GJUS&T, Hisar.

Sub: Approval of scheme of examination & syllabi of various B.Tech. programme(s) being run in University Teaching Departments as well as affiliated Engineering College(s)/Institute(s).

AND

Recommendations of Faculty of Engineering & Technology regarding Open Elective, Format of Minor Question Paper, MOOC Courses, minimum strength for Programme Elective, Semester Registration etc.

Sir,

I am directed to inform you that the Vice-Chancellor, on the recommendations of the Faculty of Engineering & Technology, vide resolutions no. 2 to 13 in its meeting held on 18.07.2019, is pleased to approve the following scheme & syllabi of B.Tech. programme(s) w.e.f. the academic session / batch mentioned against each being run in University Teaching Departments as well as affiliated colleges/institutions and recommendations of Faculty of Engineering & Technology, regarding Open Elective, format of Minor Question Paper, MOOC Courses, minimum strength for Programme Elective, Semester Registration etc. under Section 11(5) in anticipation of approval of the Academic Council of the University Act, 1995:-

1. B.Tech. (Printing Technology), B.Tech (Packaging Technology) & B.Tech. (Printing & Packaging Technology)-4th year for University Teaching Departments and affiliated colleges for 2016-17 batch onwards.
2. B.Tech. (Printing & Packaging Technology) (Part-time)-3rd & 4th year for affiliated colleges for 2017-18 batch onwards.
3. B.Tech. (CSE) & B.Tech (IT)-2nd to 4th year for University Teaching Departments and affiliated colleges for 2018-19 onwards batch.
4. B.Tech. (ECE)- 2nd to 4th year for University Teaching Departments and affiliated colleges for 2018-19 onwards batch.
5. B.Tech. (EE)- 2nd to 4th year for University Teaching Departments and affiliated colleges for 2018-19 onwards batch.

Condt..p2/-

6. B.Tech. (Printing Technology), B.Tech (Packaging Technology) & B.Tech. (Printing & Packaging Technology)-- 2nd to 4th year and syllabi of 2nd year only for University Teaching Departments and affiliated colleges for 2018-19 onwards batch.

7. B.Tech. (Agricultural Engg.) & B.Tech (Aeronautical Engineering)- 2nd to 4th year for affiliated colleges for 2018-19 onwards batch.

8. Scheme and syllabi of B.Tech. (Mechanical Engineering)- 2nd to 4th year for University Teaching Departments and affiliated colleges for 2018-19 onwards batch.

9. B.Tech. (Civil Engg.)- 2nd to 4th year for University Teaching Departments and affiliated colleges for 2018-19 onwards batch.

10. B.Tech. (Food Technology) 2nd to 4th year for University Teaching Departments for 2018-19 onwards batch.

11. The list and syllabus of Open Electives and Mandatory programmes for B.Tech. courses w.e.f. 2018-19 batch onwards.

12. The format of Minor Question paper for students of 2018-19 batch onwards (B.Tech 2nd to Final Year) and level of Assignment in light of Outcome based Education.

13. The students of B.Tech 2018-19 batch (2nd to Final year) will have choice to opt for MOOC course (not more than one in each semester) which he/she had not studied earlier, in lieu of Courses mentioned in every Programme Electives of equal credit with the prior approval of Chairperson within 15 days of start of semester.

14. The minimum 30% of existing class strength should opt for any Programme elective. Decimal part will be truncated in case 30% is not whole number.

15. The Semester Registration process being followed in FET. Faculty appreciated the process of Semester Registration in FET as it has helped in timely start of classes and has improved attendance in classes. Faculty further recommended that student should not be allowed to register after passage of one month of start of classes.

A copy of the scheme and syllabi of aforementioned B.Tech. programmes and other recommendations of Faculty of Engineering and Technology are enclosed herewith.

This is for your information and further necessary action at your end.

DA: As above

20/8/19
Deputy Registrar (Academic)
For Registrar

Endst. No.Acad./ AC-III//Fac-1 Vol.3/ 4200-4211 Dated: 20/8/19

A copy of the above is forwarded to the following for information and necessary action:-

1. Dean Academic Affairs, GJUS&T, Hisar.
2. Dean of Colleges, GJUS&T, Hisar.

contd.p/3

3. Dean, Faculty of Engg. & Technology, GJUS&T, Hisar (along with copy of scheme & syllabi of B.Tech. (Agricultural Engg.) & B.Tech (Aeronautical Engineering) 2nd to 4th year) and other recommendations from sr. no. 11 to 15. Further, he is requested to get upload the syllabi of above said programme on the website of the University.

4. Chairperson, Department of CSE, GJUS&T, Hisar (alongwith copy of scheme & syllabi of B.Tech. (CSE) & B.Tech (IT)-- 2nd to 4th year and other recommendations from sr. no. 11 to 15. Further, he is requested to get upload the syllabi of above said programme on the website of the University.

5. Chairperson, Department of Food Tech., GJUS&T, Hisar (alongwith copy of scheme & syllabi of B.Tech. (Food Technology) 2nd to 4th year and other recommendations from sr. no. 11 to 15. Further, she is requested to get upload the syllabi of above said programme on the website of the University.

6. Chairperson, Department of Mech. Engg., GJUS&T, Hisar (alongwith copy of scheme & syllabi of B.Tech. (Mechanical Engineering) 2nd to 4th year and other recommendations from sr. no. 11 to 15 . Further, he is requested to get upload the syllabi of above said programme on the website of the University.

7. Chairperson, Department of Environmental Science & Engineering, GJUS&T, Hisar (alongwith copy of scheme & syllabi of B.Tech. (Civil Engg.) 2nd to 4th year and other recommendations from sr. no. 11 to 15. Further, he is requested to get upload the syllabi of above said programme on the website of the University.

8. Chairperson, Department of Electronics & Communication Engineering GJUS&T, Hisar (alongwith copy of scheme & syllabi of B.Tech. (ECE) 2nd to 4th year and B.Tech. (EE) 2nd to 4th year and other recommendations from sr. no. 11 to 15. Further, he is requested to get upload the syllabi of above said programme on the website of the University.

9. Chairperson, Department of Printing Tech., GJUS&T, Hisar (alongwith copy of scheme & syllabi of B.Tech. (Printing Technology), B.Tech (Packaging Technology) & B.Tech. (Printing & Packaging Technology)- 4th year, B.Tech. (Printing & Packaging Technology) (Part-time) 3rd & 4th year and Scheme of examination -- 2nd to 4th year and syllabi – 2nd year of B.Tech. (Printing Technology), B.Tech (Packaging Technology) & B.Tech. (Printing & Packaging Technology) and other recommendations from sr. no. 11 to 15. Further, he is requested to get upload the syllabi of above said programme on the website of the University.

10. Director-Principal, Manav Institute of Tech. & Mgt., Village- Jevra, alongwith copy of scheme & syllabi of B.Tech. (CSE), (ECE), (ME), (Civil Engg.), (EE), (Aeronautical Engg.) and (Agriculture Engg.) programmes and other recommendations from sr. no. 11 to 15.

11.. Director-Principal, Om Institute of Tech. & Mgt., Juglan alongwith copy of scheme & syllabi of B.Tech. (CSE), (IT), (ME), (ECE), (Civil Engg.), (Printing &Packaging Technology), (Printing &Packaging Technology) – Part Time (EE), programmes and other recommendations from sr. no. 11 to 15.

20/8/19

Deputy Registrar (Academic)
for Registrar

Preface

The overall well-being of a nation depends on the eminence of its human resource. Providing quality education plays a vital role in transforming people into valuable human resource. Well educated students of today will become innovators and leaders of tomorrow who are going to ensure a constructively competitive but sustainable and peaceful world for everyone. To meet this end, AICTE has developed a model curriculum for Engineering graduates. The AICTE model curriculum is no way an ideal document and lacks quality it ought to have. The model curriculum has no uniformity from one course to another. The Course Outcomes are ill-defined, and, references are incomplete and inconsistently formatted in the model curriculum, nonetheless it has given us an opportunity to revise the curriculum of our graduate engineering programmes.

We have revised the curriculum for Bachelor of Technology Programme in Computer Science and Engineering of the Department of Computer Science and Engineering, Guru Jambheshwar University of Science and Technology, Hisar. The curriculum is designed around the framework of outcome-based education in which students are at the centre of teaching learning process. The salient features of the curriculum design are as follows:

1. To start with, four Programme Educational Outcomes are defined.
2. The twelve Programme Outcomes (POs) are taken from the Self Appraisal Report format of National Board of Accreditation (NBA) for undergraduate engineering programmes and three Programme Specific Outcomes (PSOs) are outlined to capture the specialisations of the B. Tech. (CSE) programme.
3. An induction programme of three weeks duration has been introduced to make the lately admitted students comfortable in their new environment. The induction programme continues in the form of participation in Sports club or Green club or Cultural, Literature and Film Club etc. for the remaining period of the programme. It is mandatory for every student to join in one of these clubs.
4. In addition to the professional core and elective courses, there is a provision for many courses from Basic Sciences, Engineering Sciences, Mathematics and Humanities. The non-credit mandatory courses are included to make students aware about constitution of India, issues related to environmental and sustainable development, and Indian traditional wisdom.
5. For every course, 4 to 6 Course Outcomes (COs) are defined which are concrete and measurable.
6. Guidelines for preparing sessional examination question papers and assignments have been framed for measuring the attainment levels of COs.
7. The internal and external evaluation criteria for various courses has been succinctly described.
8. The Course Outcomes (COs) are mapped to Programme Outcomes (POs) by defining a CO:PO articulation matrix for every course.
9. The methodology for computing the attainment levels for the Course Outcomes and Programme Outcomes is laid out.
10. The new curriculum has a focus on the problem solving and learning capabilities of the students. There are many laboratory courses which give students a hands-on experience in problem solving. Further, provisions for industry internship/training and project works make students ready to accept challenges and do research to solve difficult engineering problems.
11. Overall, the new curriculum is made keeping in the view the continuous cycle of improvement in teaching learning process of outcome-based education strategy.

Syllabus Revision Committee: Team Members

1. Professor Saroj (Convener)
2. Dr. Ritu Nagpal, Associate Professor, (Member)
3. Dr. Sunita Beniwal, Assistant Professor), (Member)
4. Mr. Manoj, Assistant Professor (Member)
5. Dr. Jyoti Vashishtha
6. Dr. Anju Sangwan

In addition, all the faculty members of the Department have participated in revising the syllabus of the various courses.

Chapter 1: General Information

1.1. Vision and Mission of the Department of Computer Science and Engineering

1.1.1 Vision

The vision of the Department is to become a centre of excellence for education in Computer Science and Engineering, Information Technology and Computer Applications. We visualize ourselves as an agency to nurture young minds to be the future leaders in the field of higher education, research and development, and information technology industry. Our aim is to bring out creators and innovators who will work towards the overall well-being of the society.

1.1.2 Mission

- Imparting state-of-the-art knowledge in Computer Science and Engineering, Information Technology and Computer Applications.
- Ensuring that our students graduate with a sound theoretical basis and wide-ranging practical experience.
- Fostering linkages between the Department and, public and private sectors, traversing research establishments as well as Information Technology industry.
- Promoting ethical research of high quality.
- Adopting the best pedagogical methods in order to maximize knowledge transfer.
- Inculcating a culture of free and open discussions in the Department.
- Engaging students in evolving original ideas and applying them to solve complex engineering problems.
- Inspiring a zest into students for lifelong learning.
- Infusing scientific temper, enthusiasm, professionalism, team spirit and leadership qualities in students.
- Sensitizing students to look for environmentally sustainable engineering solutions.
- Upholding democratic values and an environment of equal opportunity for everyone.

1.2 B. Tech. (CSE): Programme Educational Objectives (PEOs)

The Programme Educational Objectives of the B. Tech. (CSE) Programme are:

PEO1. To prepare responsible and ethical professionals to be successfully employed in Computer Science and Information Technology industry, who will be able to apply the principles of science, engineering and project management to develop and deploy solutions for real world problems after assessing their environmental, cultural and societal implications.

PEO2. To train students for analysing, evaluating and designing complex engineering solutions individually or in teams by doing a systematic and in-depth research in the related problem domains, by using modern tools and by communicating effectively among the various stake holders.

PEO3. To groom the professionals and entrepreneurs of tomorrow with leadership qualities and deep societal concerns who can move up in their professional career or start their own ventures.

PEO4. To guide the graduates to develop a positive attitude towards learning and motivate them to take up higher studies and research.

1.3 B. Tech. (CSE): Programme Outcomes (POs)

PO1. **Engineering knowledge**: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.

PO2. **Problem analysis**: Identify, formulate, review research literature, and analyse complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.

PO3. **Design/development of solutions**: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.

PO4. **Conduct investigations of complex problems**: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

PO5. **Modern tool usage**: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modelling to complex engineering activities with an understanding of the limitations.

PO6. **The engineer and society**: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

PO7. **Environment and sustainability**: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.

PO8. **Ethics**: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

PO9. **Individual and team work**: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.

PO10. **Communication**: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

PO11. **Project management and finance**: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

PO12. **Life-long learning**: Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

Program Specific Outcomes (PSOs)

PSO1 **Developing Computational Systems**: Use principles of electronics and Micro-Processors, various programming languages, data structures, database management systems, computer algorithms, theory of computation and software engineering for designing and implementing computational systems.

PSO2 **Devising Networking Solutions**: Apply the knowledge of systems in the areas related to network technologies, mobile ad hoc and sensor networks, cloud computing, IoT and, information and web security for devising networking solutions.

PSO3 **Doing Data Analytics and Designing Intelligent Systems**: Utilize the approaches and tools of artificial intelligence and soft computing, data analytics and machine learning for designing and working with intelligent systems that can extract valuable information from large amount of data and learn from their environment.

Chapter 2: Programme Structure and Scheme of Examination

B. Tech. (CSE)

2.1 Programme Structure and Credit Distribution in Various Components of the Curriculum

2.1.1 Definition of a Credit

Type of Teaching Learning Activity	No. of credits
1 Hour Lecture (L) per week	1 credit
1 Hour Tutorial (T) per week	1 credit
2 Hours Practical (Lab) per week	1 credit

2.1.2 Credits for Different Curriculum Components of B. Tech. Programme

Distribution of Credits			
Sr. No.	**Category**	**Course Code**	**Credit Breakup**
1.	Humanities and Social Sciences including Management courses	HSMC	07
2.	Basic Science courses	BSC	22
3.	Engineering Science courses including workshop, drawing, basics of electrical/mechanical/computer etc. including labs.	ESC	16
4.	Professional core courses +Professional Core Lab. Courses	PCC-CSE	76
5.	Professional Elective courses relevant to chosen specialization/branch	PEC-CSE	18
6.	Open subjects – Electives from other technical and /or emerging subjects	OEC	09
7.	Project work, seminar and internship in industry or elsewhere	PROJ-CSE	12
8.	Mandatory Courses [Environmental Sciences, Induction Program, Indian Constitution, Essence of Indian Knowledge Tradition]	MC	-
9.	Lab. Courses	PCC-CSE-P/ PEC-CSE-P	
Total Credits			**160**

2.1.3 Mandatory Courses (MC)

Sr. No.	Course Code	Nomenclature of Course	Hours per week			Total Credits	Semester
			L	T	P		
1.	MC101-T	Induction Training	0	0	0	0	I
2.	MC103-T	Indian Constitution	3	0	0	0	II
3.	MC102-T	Environmental Sciences	3	0	0	0	III
4.	MC104-T	Essence of Indian Traditional Knowledge	3	0	0	0	V
Total Credits						**0**	

2.1.4 Humanities and Social Sciences Including Management Courses (HSMC)

Sr. No.	Course Code	Nomenclature of Course	Hours per week			Total Credits	Semester
			L	T	P		
1.	HSMC101-T	English	2	0	2	3	II
2.	HSMC301-T	Humanities-I (Economics for Engineers)	2	0	0	2	V
3.	HSMC302-T	Humanities-II (Fundamentals of Management for Engineers)	2	0	0	2	VI
Total Credits						**7**	

2.1.5 Basic Science Courses (BSC)

Sr. No.	Code No.	Nomenclature of Course	Hours per week			Total Credits	Semester
			L	T	P		
1.	BSC101-T BSC101-P	Physics	3	1	3	5.5	II
2.	BSC105-T	Mathematics-I	3	1	0	4.0	I
3.	BSC102-T BSC102-P	Chemistry-I	3	1	3	5.5	I
4.	BSC106-T	Mathematics-II	3	1	0	4.0	II
5.	BSC201-T	Mathematics-III	3	0	0	3.0	III
Total Credits						**22**	

2.1.6 Engineering Science Courses (ESC)

Sr. No.	Course Code	Nomenclature of Course	Hours per week			Total Credits	Semester
			L	T	P		
1.	ESC101-(T/P)	Basic Electrical Engineering	3	1	2	5	II
2.	ESC102-(T/P)	Engineering Graphics & Design	1	0	4	3	I
3.	ESC103-(T/P)	Programming for problem Solving	3	0	4	5	I
4.	ESC104-(T/P)	Workshop/Manufacturing Practices	1	0	4	3	II
Total Credits						**16**	

2.1.7 Professional Core Courses (PCC-CSE)

Sr. No.	Course Code	Nomenclature of Course	Hours per week			Total Credits	Semester
			L	T	P		
1.	PCC-CSE201-(T/P)	Data Structures and Algorithms	3	0	4	5	III
2.	PCC-CSE202-(T/P)	Object Oriented Programming using C++	3	0	4	5	III
3.	PCC-CSE203-T	Discrete Mathematics	3	0	0	3	III
4.	PCC-CSE204-T	Computer Organisation and Architecture	3	0	0	3	III
5.	PCC-CSE205-(T/P)	Microprocessor and Interfacing	3	0	2	4	IV
6.	PCC-CSE206-(T/P)	Computer Networks	3	0	2	4	IV
7.	PCC-CSE207-(T/P)	Database Management System	3	0	2	4	IV
8.	PCC-CSE208-T	Analysis and Design of Algorithms	3	0	0	3	IV
9.	PCC-CSE209-T	Software Engineering	3	0	0	3	IV
10.	PCC-CSE210-(T/P)	Java Programming	3	0	4	5	IV
11.	PCC-CSE301-(T/P)	Computer Graphics	3	0	2	4	V
12.	PCC-CSE302-(T/P)	Python Programming	3	0	3	4.5	V
13.	PCC-CSE303-T	High Speed Networks	3	0	0	3	V
14..	PCC-CSE304-T	Cryptography and Network Security	3	0	0	3	V
15.	PCC-CSE305-(T/P)	Operating Systems	3	0	2	4	VI
16.	PCC-CSE306-T	Formal Language and Automata Theory	3	0	0	3	VI
17.	PCC-CSE307-T	Data Analytics using R	2	0	3	3.5	VI
18.	PCC-CSE308-(T/P)	.NET using C#	2	0	2	3.0	VI
19.	PCC-CSE401-T	Compiler Design	3	0	0	3	VII
20.	PCC-CSE402-T	Artificial Intelligence	3	0	0	3	VII
21	PCC-CSE403-T	Data Mining Techniques	3	0	0	3	VIII
Total Credits						**76**	

2.1.8 Professional Elective Courses (PEC-CSE)

Sr. No.	Course Code	Nomenclature of Course	Hours per week			Total Credits	Semester
			L	T	P		
1.	PEC-CSE301-T to PEC-CSE604-T	Embedded System Design, Soft Computing, Graph Theory, Bioinformatics	3	0	0	3	VI
2.	PEC-CSE401-T to PEC-CSE404-T	Software Project Management, Wireless and Mobile Communications, Distributed Operating Systems, Cloud Computing	3	0	0	3	VII
3.	PEC-CSE405-(T/P) to PEC-CSE408-(T/P)	Advanced Microprocessor, Mobile Application Development, Multimedia Technologies, Digital Image Processing	3	0	1	4	VII
4.	PEC-CSE409-(T/P) to PEC-CSE412-(T/P)	Internet of Things, Software Defined Networks, Network Administration and Management, Software Testing and Quality Assurance	3	0	1	4	VIII
5.	PEC-CSE413-(T/P) to PEC-CSE417-(T/P)	Machine Learning, Big Data Analytics, Web Development, Statistical Computing, Digital Forensics	3	0	1	4	VIII
Total Credits						**18**	

2.1.9 Open Elective Courses (OEC)

Sr. No.	Course Code	Nomenclature of Course	Hours per week			Total Credits	Semester
			L	T	P		
1.	OEC-I	Open Elective-I	3	0	0	3	V
2.	OEC-II	Open Elective-II	3	0	0	3	VI
3.	OEC-III	Open Elective-III	3	0	0	3	VII
Total Credits						**9**	

2.1.10 Project work

Sr. No.	Course Code	Nomenclature of Course	Hours per week			Total Credits	Semester
			L	T	P		
1	INT-CSE301	Industrial Training/Internship	0	0	2	1	After IVth Sem.
2	PROJ-CSE402	Mini-Project using open source tools	0	0	1	1	After VIth Sem.
3	PROJ-CSE401	Major Project I	0	0	8	4	VII
4	PROJ-CSE403	Major Project II	0	0	12	6	VIII
Total Credits						**12**	

2.2. B. Tech. (Computer Science and Engineering): Semester-wise Scheme

Induction Programme

It is mandatory to conduct an induction programme for newly admitted students right at the beginning of the first semester. The objective of the induction programme is to create a bond between the institution and the newly admitted students.

The new students enter an institution with diverse backgrounds and expectations. It is important to help them adjust to the new environment. To meet this purpose, there will be three week-long induction programme before the normal classes start. The induction program shall provide students the opportunity to settle down and be comfortable in the new environment. The new students will come to know their seniors, faculty members, department and university. The student would be engaged in the following activities.

1. Familiarization with the Department and the University
2. Physical activities like morning walks, cycling or playing one or the other games.
3. Creative arts like painting, music and dancing etc.
4. Talks and lectures by eminent people, and group discussion on universal Human values
5. Literary activities like reading writing or debating

The schedule for organizing the induction programme shall be prepared every year at university level.

Students will be engaged in diverse activities at the level of Department. Depending on the interest, every student must opt for one of the activities during all the semesters. For this purpose, the following clubs shall be established in the Department.

1. Sports Club
2. Green Club
3. Culture, Literature and Film Club
4. Social Service Club
5. Technology Innovation Club

Each student will spend 3 to 5 hours for these activities per week.

SEMESTER I

Sr. No.	Course Code	Nomenclature of the Course	Hours per week			Credits
			L	T	P	
1.	BSC101-T BSC101-P	Physics (Group A)	3	1	3	5.5
	BSC102-T BSC102-P	Chemistry (Group B)				
2.	BSC103-T BSC105-T	Mathematics –I Mathematics –I (for CSE/IT)	3	0	1	4.0
3.	ESC101-T ESC101-P	Basic Electrical Engineering (Group A)	3	1	2	5.0
	ESC103-T ESC103-P	Programming for Problem Solving (Group B)	3	0	4	
4.	ESC104-T ESC104-P	Workshop/Manufacturing Practices (Group A)	1	0	4	3
	ESC102-T ESC102-P	Engineering Graphics & Design (Group B)	1	0	4	
5.	MC101	Induction Training (Group A & B)	3 weeks	0	0	0
Total Credits						17.5

SEMESTER II

Sr. No.	Course Code	Nomenclature of the Course	Hours per week			Credits
			L	T	P	
1.	BSC101-T BSC101-P	Physics (Group B)	3	1	3	5.5
	BSC102-T BSC102-P	Chemistry (Group A)				
2.	BSC104-T BSC106-T	Mathematics –II Mathematics –II (for CSE/IT)	3	0	1	4.0
3.	ESC101-T ESC101-P	Basic Electrical Engineering (Group B)	3	1	2	5.0
	ESC103-T ESC103-P	Programming for Problem Solving (Group A)	3	0	4	
4.	ESC104-T ESC104-P	Workshop/Manufacturing Practices (Group B)	1	0	4	3
	ESC102-T ESC102-P	Engineering Graphics & Design (Group A)	1	0	4	
5.	HSMC101-T HSMC101-P	English (Group A and B)	2	0	2	3
6.	MC102-T	Environmental Sciences (Group A)	3	0	0	0
	MC103-T	Indian Constitution (Group B)	3	0	0	
Total Credits						20.5

SEMESTER-III

Sr. No.	Course Code	Nomenclature of the Course	Hours per week			Credits
			L	T	P	
1.	BSC201-T	Mathematics-III	3	0	0	3
2,	PCC-CSE201-T/ PCC-IT201-T	Data Structures and Algorithms	3	0	0	3
3.	PCC-CSE202-T/ PCC-IT-202-T	Object Oriented Programming using C++	3	0	0	3
4.	PCC-CSE203-T/ PCC-IT203-T	Discrete Mathematics	3	0	0	3
5.	PCC-CSE204-T/ PCC-IT204-T	Computer Organisation and Architecture	3	0	0	3
6.	MC102-T	Environmental Science	3	0	0	0
7.	PCC-CSE201-P/ PCC-IT201-P	Data Structures and Algorithms using C/C++ Lab.	0	0	4	2
8.	PCC-CSE202-P/ PCC-IT202-P	Object Oriented Programming using C++ Lab.	0	0	4	2
Total Credits						19

SEMESTER IV

Sr. No.	Course Code	Nomenclature of the Course	Hours per week			Credits
			L	T	P	
1.	PCC-CSE205-T/ PCC-IT205-T	Microprocessor and Interfacing	3	0	0	3
2.	PCC-CSE206-T/ PCC-IT301-T	Computer Networks	3	0	0	3
3.	PCC-CSE207-T/ PCC-IT207-T	Database Management System	3	0	0	3
4.	PCC-CSE208-T/ PCC-IT208-T	Analysis and Design of Algorithms	3	0	0	3
5.	PCC-CSE209-T/ PCC-IT209-T	Software Engineering	3	0	0	3
6.	PCC-CSE210-T/ PCC-IT210-T	Java Programming	3	0	0	3
7.	PCC-CSE205-P PCC-IT205-P	Microprocessor and Interfacing Lab.	0	0	2	1
8.	PCC-CSE206-P/ PCC-IT301-P	Computer Networks Lab.	0	0	2	1
9.	PCC-CSE207-P PCC-IT207-P	Database Management System Lab.	0	0	2	1
10.	PCC-CSE210-P/ PCC-IT210-P	Java Programming Lab.	0	0	4	2
Total Credit						23
Industrial Training of 4-6 weeks after IVth semester						

Semester V

Sr. No.	Course Codes	Nomenclature of the Course	Hours per week			Credits
			L	T	P	
1.	PCC-CSE301-T/ PEC-IT409-T	Computer Graphics	3	0	0	3
2.	PCC-CSE302-T/ PEC-IT308-T	Python Programming	3	0	0	3
3.	PCC-CSE303-T/ PEC-IT305-T	High Speed Network Technologies	3	0	0	3
4.	PCC-CSE304-T/ PEC-IT-402-T	Cryptography and Network Security	3	0	0	3
5.	OEC-I	Open Elective Course offered by other Departments	3	0	0	3
6.	HSMC301-T	Economics for Engineers	2	0	0	2
7.	MC104-T	Essence of Indian Traditional Knowledge	3	0	0	0
8.	PCC-CSE301-P/ PEC-IT-409-P	Computer Graphics Lab.	0	0	2	1
9.	PCC-CSE302-P/ PCC-IT308-P	Python Programming Lab.	0	0	3	1.5
10.	INT-CSE301	Industrial Training/Internship	0	0	0	1
Total Credits						20.5

SEMESTER VI

Sr. No.	Course Codes	Nomenclature of the Course	Hours per week			Credits
			L	T	P	
1.	PCC-CSE305-T/ PCC-IT206-T	Operating Systems	3	0	0	3
2.	PCC-CSE306-T/ PCC-IT303-T	Formal Language and Automata Theory	3	0	0	3
3.	PCC-CSE307-T/ PEC-IT407-T	Data Analytics using R	2	0	0	2
4.	PCC-CSE308-T/ PCC-IT302-T	.NET using C#	2	0	0	2
5.	PEC-CSE301-T to PEC-CSE304-T	Professional Elective Course to be opted by students	3	0	0	3
6.	HSMC302-T	Fundamentals of Management for Engineers	2	0	0	2
7.	OEC-II	Open Elective Course offered by other Departments	3	0	0	3
8.	PCC-CSE305-P/ PCC-IT206-P	Operating Systems Lab. (UNIX/LINUX)	0	0	2	1
9.	PCC-CSE307-P/ PEC-IT407-P	Data Analytics using R Lab.	0	0	3	1.5
10.	PCC-CSE308-P/ PCC-IT302-P	.NET using C# Lab.	0	0	2	1
Total Credits						21.5
A Mini-Project/Training based on open source tools/.NET						

List of Electives I

1. PEC-CSE301-T/ PEC-IT301-T: Embedded System Design
2. PEC-CSE302-T/ PCC-IT401-T: Wireless and Mobile Communications
3. PEC-CSE303-T/ PEC-IT303-T: Graph Theory
4. PEC-CSE304-T/ PEC-IT304-T: Bioinformatics
5. Any one of the MOOC not studied earlier and Sangit of equal credits.

SEMESTER VII

Sr. No.	Course Codes	Nomenclature of the Course	Hours per week			Credits
			L	T	P	
1.	PCC-CSE401-T/ PCC-IT306-T	Compiler Design	3	0	0	3
2.	PCC-CSE402-T/ PCC-IT304-T	Artificial Intelligence	3	0	0	3
3.	PEC-CSE401-T to PEC-CSE404-T	Professional Elective Course to be opted by students	3	0	0	3
4.	PEC-CSE405-T to PEC-CSE408-T	Professional Elective Course to be opted by students	3	0	0	3
5.	OEC-III	Open Elective Course offered by other Departments	3	0	0	3
6.	PEC-CSE405-P to PEC-CSE 408-P	Professional Elective Course Lab.	0	0	2	1
7.	PROJ-CSE401	Major Project I	0	0	8	4
8.	PROJ-CSE402	Mini Project using open source tools/.NET	0	0	2	1
Total Credits						21

List of Electives II

1. PEC-CSE401-T/ PEC-IT401-T: Software Project Management
2. PEC-CSE302-T/ PEC-IT302-T: Soft Computing
3. PEC-CSE403-T/ PEC-IT403-T: Distributed Operating Systems
4. PEC-CSE404-T/ PEC-IT404-T: Cloud Computing
5. Any one of the MOOC not studied earlier and of equal credits.

List of Elective III

1. PEC-CSE405-T/ PEC-IT405-T: Advanced Microprocessor
2. PEC-CSE406-T/ PCC-IT403-T: Mobile Application Development
3. PEC-CSE407-T/ PEC-IT411-T: Multimedia Technologies
4. PEC-CSE408-T/ PEC-IT408-T: Digital Image Processing
5. Any one of the MOOC not studied earlier and of equal credits.

List of Elective III (Labs.)

1. PEC-CSE405-P/ PEC-IT405-P: Advanced Microprocessor (Lab.)
2. PEC-CSE406-P/ PCC-IT403-P: Mobile Application Development (Lab.)
3. PEC-CSE407-P/ PEC-IT411-P: Multimedia Technologies (Lab.)
4. PEC-CSE408-P/ PEC-IT408-P: Digital Image Processing (Lab.)

Mathematics III

e Information

ode: BSC201-T redits: 3 sic Sciences Hours: 3hours/week ectures (L) tion Duration: 3 hours	**Course Assessment Methods (internal: 30; external: 70)** Two minor examinations (20 marks), Class Performance measured through percentage of lectures attended (4 marks), assignments (6 marks), and the end- semester examination (70 marks). For the end semester examination, nine questions are to be set by the examiner. A candidate is required to attempt 5 questions in all. All questions carry equal marks. Question number 1 will be compulsory and based on the entire syllabus. It will contain seven parts of 2 marks each. Question numbers 2 to 9 will be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt the remaining four questions by selecting one question from each of the four units.

Mathematics I and Mathematics II

rse

anced mathematics course that offers the knowledge of Fourier Series, Fourier Transforms, Complex Variables. These concepts are essential for students to solve problems in image tal signal processing and other related engineering fields.

nes: By the end of the course students will be able to:

ine concepts and terminology of Fourier Series and Fourier transforms, Functions of complex iables and Power Series etc. (LOTS: Level 1: Remember)
ve problems using Fourier transforms in domains like digital electronics and image processing. OTS: Level 3: Apply)
olyprinciples of functions of complex variables to solve computational problems.(LOTS: Level Apply)
npare various concepts related to Fourier transforms and functions of complex iables.(HOTS: Level 4: Analyse)
ect suitablemethod for given computational engineering problems and related domain.(HOTS: vel 4: Evaluate)
egrate the knowledge of Fourier Series and Fourier transforms, Functions of complex variables, l Power Series for solving real world problems. (HOTS: Level 6: Create)

Course Content

ınd Fourier Transforms: Euler's formulae, conditions for a Fourier expansion, change of interval, ion of odd and even functions, Fourier expansion of square wave, rectangular wave, saw-toothed full rectified wave, half range sine and cosine series.

ls, Fourier transforms, Shifting theorem (both on time and frequency axes), Fourier transforms of urier transforms of integrals, Convolution theorem, Fourier transform of Dirac delta function.

SEMESTER VI

Sr. No.	Course Codes	Nomenclature of the Course	Hours per week			Credits
			L	T	P	
1.	PCC-CSE305-T/ PCC-IT206-T	Operating Systems	3	0	0	3
2.	PCC-CSE306-T/ PCC-IT303-T	Formal Language and Automata Theory	3	0	0	3
3.	PCC-CSE307-T/ PEC-IT407-T	Data Analytics using R	2	0	0	2
4.	PCC-CSE308-T/ PCC-IT302-T	.NET using C#	2	0	0	2
5.	PEC-CSE301-T to PEC-CSE304-T	Professional Elective Course to be opted by students	3	0	0	3
6.	HSMC302-T	Fundamentals of Management for Engineers	2	0	0	2
7.	OEC-II	Open Elective Course offered by other Departments	3	0	0	3
8.	PCC-CSE305-P/ PCC-IT206-P	Operating Systems Lab. (UNIX/LINUX)	0	0	2	1
9.	PCC-CSE307-P/ PEC-IT407-P	Data Analytics using R Lab.	0	0	3	1.5
10.	PCC-CSE308-P/ PCC-IT302-P	.NET using C# Lab.	0	0	2	1
Total Credits						21.5
A Mini-Project/Training based on open source tools/.NET						

List of Electives I

1. PEC-CSE301-T/ PEC-IT301-T: Embedded System Design
2. PEC-CSE302-T/ PCC-IT401-T: Wireless and Mobile Communications
3. PEC-CSE303-T/ PEC-IT303-T: Graph Theory
4. PEC-CSE304-T/ PEC-IT304-T: Bioinformatics
5. Any one of the MOOC not studied earlier and of equal credits.

SEMESTER VII

Sr. No.	Course Codes	Nomenclature of the Course	Hours per week			Credits
			L	T	P	
1.	PCC-CSE401-T/ PCC-IT306-T	Compiler Design	3	0	0	3
2.	PCC-CSE402-T/ PCC-IT304-T	Artificial Intelligence	3	0	0	3
3.	PEC-CSE401-T to PEC-CSE404-T	Professional Elective Course to be opted by students	3	0	0	3
4.	PEC-CSE405-T to PEC-CSE408-T	Professional Elective Course to be opted by students	3	0	0	3
5.	OEC-III	Open Elective Course offered by other Departments	3	0	0	3
6.	PEC-CSE405-P to PEC-CSE 408-P	Professional Elective Course Lab.	0	0	2	1
7.	PROJ-CSE401	Major Project I	0	0	8	4
8.	PROJ-CSE402	Mini Project using open source tools/.NET	0	0	2	1
Total Credits						21

List of Electives II

1. PEC-CSE401-T/ PEC-IT401-T: Software Project Management
2. PEC-CSE302-T/ PEC-IT302-T: Soft Computing
3. PEC-CSE403-T/ PEC-IT403-T: Distributed Operating Systems
4. PEC-CSE404-T/ PEC-IT404-T: Cloud Computing
5. Any one of the MOOC not studied earlier and of equal credits.

List of Elective III

1. PEC-CSE405-T/ PEC-IT405-T: Advanced Microprocessor
2. PEC-CSE406-T/ PCC-IT403-T: Mobile Application Development
3. PEC-CSE407-T/ PEC-IT411-T: Multimedia Technologies
4. PEC-CSE408-T/ PEC-IT408-T: Digital Image Processing
5. Any one of the MOOC not studied earlier and of equal credits.

List of Elective III (Labs.)

1. PEC-CSE405-P/ PEC-IT405-P: Advanced Microprocessor (Lab.)
2. PEC-CSE406-P/ PCC-IT403-P: Mobile Application Development (Lab.)
3. PEC-CSE407-P/ PEC-IT411-P: Multimedia Technologies (Lab.)
4. PEC-CSE408-P/ PEC-IT408-P: Digital Image Processing (Lab.)

SEMESTER VIII

Sr. No.	Course Codes	Nomenclature of the Course	Hours per week			Credits
			L	T	P	
1.	PCC-CSE403-T/ PCC-IT402-T	Data Mining Techniques	3	0	0	3
2.	PEC-CSE409-T to PEC-CSE412-T	Professional Elective Course to be opted by students	3	0	0	3
3.	PEC-CSE413-T to PEC-CSE417-T	Professional Elective Course to be opted by students	3	0	0	3
4.	PEC-CSE409-P to PEC-CSE412-P	Professional Elective Course Lab.	0	0	2	1
5.	PEC-CSE413-P to PEC-CSE417-P	Professional Elective Course Lab.	0	0	2	1
6.	PROJ-CSE403	Major Project II	0	0	12	6
Total Credits						17

List of Electives IV

1. PEC-CSE409-T/ PEC-IT409-T: Internet of Things
2. PEC-CSE410-T/ PEC-IT410-T: Software Defined Networks
3. PEC-CSE411-T/ PCC-IT305-T: Network Administration and Management
4. PEC-CSE412-T/ PEC-IT412-T: Software Testing and Quality Assurance
5. Any one of the MOOC not studied earlier and of equal credits

List of Electives IV (Labs.)

1. PEC-CSE409-P/ PEC-IT409-P: Internet of Things (Lab.)
2. PEC-CSE410-P/ PEC-IT410-P: Software Defined Networks (Lab.)
3. PEC-CSE411-P/ PCC-IT305-P: Network Administration and Management(Lab.)
4. PEC-CSE412-P/ PEC-IT412-P: Software Testing and Quality Assurance (Lab.)

List of Electives V

1. PEC-CSE413-T/ PEC-IT413-T: Machine Learning
2. PEC-CSE414-T/ PEC-IT414-T: Big Data Analytics
3. PEC-CSE415-T/ PEC-IT415-T: Web Development
4. PEC-CSE416-T/ PEC-IT416-T: Statistical Computing
5. PEC-CSE417-T/ PEC-IT406-T: Digital Forensics
6. Any one of the MOOC not studied earlier and of equal credits.

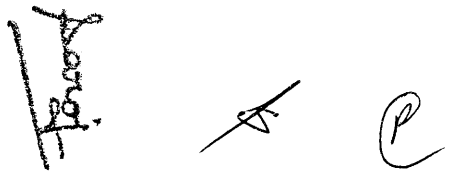
List of Electives V (Labs.)

1. PEC-CSE413-P/ PEC-IT413-P: Machine Learning (Lab.)
2. PEC-CSE414-P/ PEC-IT414-P: Big Data Analytics (Lab.)
3. PEC-CSE415-P/ PEC-IT415-P: Web Development (Lab.)
4. PEC-CSE416-P/ PEC-IT416-P: Statistical Computing (Lab.)
5. PEC-CSE417-P/ PEC-IT406-P: Digital Forensics (Lab.)

Chapter 3: Detailed Syllabi of Various Courses

CO-PO Articulation Matrix Mathematics-III (BSC201-T)

List of Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO13	PSO14	PSO15
CO1. **Define** concepts and terminology of Fourier Series and Fourier transforms, Functions of complex variables and Power Series. (LOTS: Level 1: Remember)	1	--	-		-	-	-	-	-	-	-	-	2	2	
CO2. **Solve** problems using Fourier transforms in domains like digital electronics and image processing. (LOTS: Level 3: Apply)	2	2	2	2	-	-	-	-	-	-	-	-	3	2	2
CO3. **Apply**principles of functions of complex variables to solve computational problems. (LOTS: Level 3: Apply). (LOTS: Level 3: Apply)	2	2	2	2	-	-	-	-	-	-	-	-	3	2	3
CO4. **Compare**various concepts related to Fourier transforms and functions of complex variables (HOTS: Level 4: Analyse).	3	3	2	3	-	-	-	-	-	-	-	-	3	2	3
CO5. **Select** suitablemethod for given computational engineering problems and related domain. (HOTS: Level 4: Evaluate)	3	3	2	3	-	-	-	-	-	-	-	-	3	2	3
CO6. **Integrate** the knowledge of Fourier Series and Fourier transforms, Functions of complex variables and Power Series for solving real world problems. (HOTS: Level 6: Create)	3	3	2	3	-	-	-	-	-	-	-	-	2	2	3



Data Structures and Algorithms

General Course Information

Course Code: PCC-CSE201-T/ PCC-IT201-T Course Credits: 3 Type: Professional Core Contact Hours: 3hours/week Mode: Lectures (L) Examination Duration: 3 hours	**Course Assessment Methods (internal: 30; external: 70)** Two minor examinations (20 marks), Class Performance measured through percentage of lectures attended (4 marks), assignments (6 marks), and the end- semester examination (70 marks). For the end semester examination, nine questions are to be set by the examiner. A candidate is required to attempt 5 questions in all. All questions carry equal marks. Question number 1 will be compulsory and based on the entire syllabus. It will contain seven parts of 2 marks each. Question numbers 2 to 9 will be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt the remaining four questions by selecting one question from each of the four units.

Pre-requisites: Programming in C

About the Course:

Data Structure and Algorithms is a core and an essential course for every graduate in Computer Science and Engineering. This course introduces data structures like arrays, linked lists, trees and graphs etc. and various operations to be implemented on these data structures for solving real world problems. It includes various sorting and searching algorithms as well. Further, it incorporates complexity analysis of algorithms implemented on various data structures.

Course Outcomes: By the end of the course students will be able to:

CO1. **describe** various types of data structures and operations that can be implemented on these data structures. (LOTS: Level 1: Remember)
CO2. **demonstrate** the use of various data structures and their related operations. (LOTS: Level 2: Understand)
CO3. **apply** data structure to solve computational problems. (LOTS: Level 3: Apply)
CO4. **compare** the suitability of alternative data structures and prescribed operations for various problem situations. (HOTS: Level 4: Analyse).
CO5. **defend** solutions with respect to effective storage of data and efficiency of the required operations for solving real world problems. (HOTS: Level 5: Evaluate)

Course Content

Unit I

Introduction to data structures and their types, Abstract data types, Linear lists: Arrays and linked lists: memory representations, implementing operations like traversing, searching, inserting and deleting etc. Applications of arrays and linked lists. Representing sets and polynomials using linked lists.

Unit II

Stack and Queue: Static and linked implementations, Operations and Applications. Circular queues, Tress, Binary trees and related terminology, Tree traversals (Recursive), Threaded Binary Trees, Binary Search Trees implementation and operations, Priority queues.

Unit III

Height Balanced or AVL trees and B trees. Graph definitions and related terminology, memory representations and related operations (traversal, insertion, deletion, search), Path Matrix, Warshall's Shortest path algorithm Hashing, Hash tables, hash function and collision resolution.

Unit IV

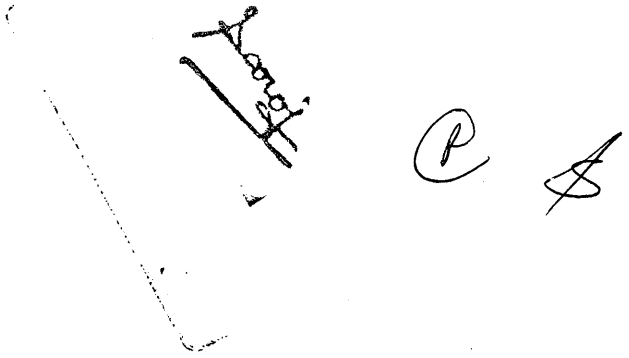
Sequential and binary search, Sorting algorithms: Bubble sort, Selection sort, Insertion sort, Quick sort, Merge sort, Count sort, Heap sort, Comparison of searching and sorting techniques based on their complexity analysis, Time and space complexity of algorithms: Asymptotic analysis, Big O, Omega, Theta notations.

Text and Reference Books:

1. Aho, A. V., Ullman, J. D., and Hopcroft, J. E., *Data Structures and Algorithms*, Addison-Wesley, 1983.
2. LangsamYedidyah, Augenstein J Moshe, Tenenbaum M Aaron, *Data Structures using C and C++*, 3rd edition, PHI, 2009.
3. Cormen, T. H., Leiserson, C. E., Rivest, R. L. and Stein, C., *Introduction to Algorithms*, MIT Press, 2009.
4. Robert L. Kruse, *Data Structure and Program Design in C*, Pearson Education India, 2007.
5. Weiss, M. A., *Data Structures and Algorithm Analysis in C++*, Addison-Wesley, 2007.
6. Sahni, S., *Data Structures, Algorithms, and Applications in C++*, WCB/McGraw-Hill, 2001.

CO-PO Articulation Matrix Data Structures and Algorithms Course (PCC-CSE201-T)

Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1. **Describe** various types of data structures and operations that can be implemented on these data structures. (LOTS: Level 1: Remember)	1	-	-	-	-	-	-	-	-	-	-	-	3	-	-
CO2. **Demonstrate** the use of various data structures and their related operations. (LOTS: Level 2: Understand)	1	-	-	-	-	-	-	-	-	-	-	-	3	-	-
CO3. **Apply** data structure to solve computational problems. (LOTS: Level 3: Apply)	2	2	-	-	2	-	-	-	-	-	-	-	3	2	2
CO4. **Compare** the suitability of alternative data structures and prescribed operations for solving a problem. (HOTS: Level 4: Analyse).	2	2	-	-	-	-	-		-	-	-	-	3	2	2
CO5. **Defend** solutions with respect to effective storage of data and efficiency of the required operations for solving computational problems. (HOTS: Level 5: - Evaluate)	3	3	-	1	-	-	-	-	-	-	-	-	3	2	2
Level of Attainments PCC-CSE201-T															



Object Oriented Programming using C++

General Course Information

Course Code: PCC-CSE202-T / PCC-IT202-T Course Credits: 3 Type: Professional Core Contact Hours: 3 hours/week Mode: Lectures (L) Examination Duration: 3 hours	**Course Assessment Methods (internal: 30; external: 70)** Two minor examinations (20 marks), Class Performance measured through percentage of lectures attended (4 marks), assignments (6 marks), and the end- semester examination (70 marks). For the end semester examination, nine questions are to be set by the examiner. A candidate is required to attempt 5 questions in all. All questions carry equal marks. Question number 1 will be compulsory and based on the entire syllabus. It will contain seven parts of 2 marks each. Question numbers 2 to 9 will be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt the remaining four questions by selecting one question from each of the four units.

Pre-requisites: Knowledge of computer fundamentals and problem solving using C programming

About the Course:

Objected Oriented Programming using C++ is an essential course for every graduate in Computer Science and Engineering. This course introduces the Object Oriented concepts such as data encapsulation, data hiding, data abstraction, reusability, exception handling etc., and their implementation using C++.

Course Outcomes: By the end of the course students will be able to:

CO1. **List** the concepts related to object oriented paradigms. (LOTS: Level 1: Remember)
CO2. **Distinguish** between structured and object oriented approaches to programming. (LOTS: Level 2: Understand)
CO3. **Apply** object oriented constructs for problem solving. (LOTS: Level 3: Apply)
CO4. **Detect** logical and run time errors and suggest appropriate modifications. (HOTS: Level 4: Analyse)
CO5. **Justify** the design of a program for a given problem. (HOTS: Level 5: Evaluate)
CO6. **Design** solutions to programming problems using multiple object oriented programming constructs together. (HOTS: Level 6: Create)

Course Content

Unit I

Introduction to object oriented programming, C++ standard library, basics of a typical C++ environment, illustrative simple C++ programs, new features of ANSI C++ standard, OOPs concepts: Information hiding, encapsulation, data abstraction, access modifiers, controlling access to a class level, method, or variable (public, protected, private, block level, scope and mutable), other modifiers. Structure of class and struct in memory, accessing members of structures, Class scope and accessing class members, separating interface from implementation, pre-processors directives, macro programs, header files and namespaces, default constructors, chained constructor, default arguments with constructors, constant object and const member functions, object as member of class, use of destructors, virtual destructors, controlling access function and utility functions, function overloading.

Unit II

Inline function, friend function and friend classes, using this pointer, dynamic memory allocation with new and delete, static class members, proxy class, polymorphism concepts, overloading, overriding methods, abstract classes, reusability, class's behaviors, inheritance, base classes and derived classes, protected members, casting base-class pointers to derived-class pointers, using member functions, overriding base-class members in a derived-class, public, protected and private inheritance, using constructors and destructors in derived classes, implicit derived-class object to base- class object conversion, composition vs. inheritance.

Unit III

Virtual functions, abstract base classes and concrete classes, new classes and dynamic binding, virtual destructors, fundamentals of operator overloading, restrictions on operators overloading, operator functions as class members vs. as friend functions, overloading, <<, >> overloading unary operators, overloading binary operators. I/O Streams, files handling, creating a sequential access file, reading data from a sequential access file, updating sequential access files, random access files, creating a random access file, writing data randomly to a random access file, reading data sequentially from a random access file.

Unit IV

Managing Console I/O, stream input/output classes and objects, stream output, stream input, unformatted I/O (with read and write), stream manipulators, stream format states, stream error states, exception handling, basics of C++ exception handling(try, throw, catch), rethrowing an exception, specific exception, processing unexpected exceptions, stack unwinding, exception handling in constructors and destructors, inheritance with exception introduction to generic classes, function templates, overloading template functions, class template, non-type parameters, templates and inheritance, templates and friends, templates and static members, container, iterator, algorithm and functional classes.

Text and Reference Books:

1. H. M. Deitel and P. J. Deitel, *C++ How To Program*, 6th Ed., Prentice Hall, 2008.
2. Robert Lafore, *Object-Oriented Programming in C++*, 3rd Ed., Sams Publishing, 2001.
3. D. Ravichandran, *Programming with C++*, 3rd Ed., T.M.H, 2011.
4. E. Balagurusamy, *Object oriented Programming with C++*, 6th Ed., Tata McGraw-Hill, 2013.
5. Horstmann, *Computing Concepts with C++ Essentials*, 3rd Ed., John Wiley, 2003.
6. Herbert Schildt , *The Complete Reference in C++*, 5th Ed., TMH, 2012.

CO-PO Articulation Matrix Object Oriented Programming Using C++ Course (PCC-CSE202-T)

Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1. **list** the concepts related to object oriented paradigms. (LOTS: Level 1: Remember)	1	1	-	-	-	-	-	-	-	-	-	-	2	-	-
CO2. **distinguish** between structured and object oriented approaches to programming. CO3. (LOTS: Level 2: Understand)	1	1	-	-	-	-	-	-	-	-	-	-	3	-	-
CO4. **Apply** object oriented constructs for problem solving. (LOTS: Level 3: Apply)	2	1	-	-	2	-	-	-	-	-	-	-	3	-	-
CO5. **Detect** logical and run time errors and suggest appropriate modifications. CO6. (HOTS: Level 4: Analyse)	2	2	-	-	-	-	-	-	-	-	-	-	3	-	-
CO7. **Justify** the design of a program for a given problem. (HOTS: Level 5: Evaluate)	2	3	-	-	-	-	-	-	1	-	-	-	3	-	-
CO8. **Design** solutions to programming problems using multiple object oriented programming constructs together. (HOTS: Level 6: Create)	3	3	1	-	2	-	-	-	1	-	-	-	3	-	-
Level of Attainments PCC-CSE202-T															

Discrete Mathematics

General Course Information

Course Code: PCC-CSE203-T/ PCC-IT203-T Course Credits: 3 Type: Professional Core Contact Hours: 3 hours/week Mode: Lectures (l) Examination Duration: 3 hours	**Course Assessment Methods (internal: 30; external: 70)** Two minor examinations (20 marks), Class Performance measured through percentage of lectures attended (4 marks), assignments (6 marks), and the end- semester examination (70 marks). For the end semester examination, nine questions are to be set by the examiner. A candidate is required to attempt 5 questions in all. All questions carry equal marks. Question number 1 will be compulsory and based on the entire syllabus. It will contain seven parts of 2 marks each. Question numbers 2 to 9 will be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt the remaining four questions by selecting one question from each of the four units.

Pre-requisites: Basic knowledge of Number Theory, Calculus and Algebra

About the Course:

Discrete Mathematics is a core and an essential course for every graduate in Computer Science and Engineering. This branch of mathematics mainly deals with discrete objects (as computer runs on discrete steps). It provides a mathematical language for computer science to resolve many real world problems by incorporating different methods applicable to various discrete structures. This course introduces set theory, propositional calculus, algebraic structures, recurrence relations and graph theory.

Course Outcomes: By the end of the course a student would be able to:

CO1. **outline** various discrete structures and the related operations. (LOTS: Level 1: Remember)
CO2. **illustrate** different discrete structures with the help of examples. (LOTS: Level 2: Understand)
CO3. **apply** appropriate techniques to solve problems related to discrete structures.(LOTS: Level 3: Apply)
CO4. **justify** the solutions with the help of proofs. (HOTS: Level 5: Evaluate)
CO5. **combine** techniques related to discrete structures for solving real world problems. (HOTS: Level 6: Create)

Course Content

Unit I

Set Theory: Introduction to Set Theory, Venn Diagrams, Set Operations, Algebra of Sets, Duality, Finite, Infinite Sets and Counting Principle, Classes of Sets, Power Sets, Partitions, Multi Sets, Relations: Cartesian Product, Representation of Relations, Types of Relation, Equivalence Relations and Partitions, Partial Ordering Relations, Functions: Definition, Types of Functions, Composition of Functions, Inverse Function, Recursively Defined Functions.

Unit II

Logic and Propositional Calculus: Introduction, Propositions and Compound Propositions, Basic Logical Operations, Propositions and Truth Tables, Tautologies and Contradictions, Logical Equivalence, Algebra of Propositions, Conditional and Bi-conditional Statements, Algebraic Structures: Group Axioms, Monoid, Semi-Groups, Subgroups, Abelian Group, Cosets, Normal Subgroup, Cyclic Group, Permutation Group, Lagrange's Theorem, Homomorphism, Isomorphism, Automorphism, Rings, Integral Domains and Fields (Also, some basic and standard results related to Groups, Rings, ID and Fields).

Unit III

Recursion and Recurrence Relation: Polynomials and their evaluation, Sequences, Introduction to AP, GP and AG Series, Partial Fractions, Recurrence Relation, Linear Recurrence Relations with Constant Coefficients, Linear Homogeneous Recurrence Relations with Constant Coefficients, Particular Solution- Homogeneous Linear Difference Equations, Non-Homogeneous Linear Difference Equations, Total Solution, Generating Functions.

Unit IV

Graphs Theory: Introduction to Graphs, Multi Graph, Directed and Undirected Graphs, Subgraphs, Bipartite Graphs, Regular Graphs, Connected Graphs, Homomorphic and Isomorphic Graphs, Cut points and Bridges, Paths and Circuits, Euler Graph, Hamiltonian Graph, Planar Graph, Euler Formula, Weighted Graphs, Dijkstra's Shortest Path Algorithm for Weighted Graphs, Trees, Spanning Trees, Minimum Spanning Tree (Prim's and Kruskal's Algorithm).

Text and Reference Books:

1. J.P. Trembley and R. Manohar, *Discrete Mathematical Structures with Applications to Computer Science*, Tata McGraw Hill – 13th reprint, 2012.
2. Kenneth H. Rosen, *Discrete Mathematics and its applications*, 6th Edition, Tata McGraw Hill, 2011.
3. Richard Johnsonbaugh, *Discrete Mathematics*, 6th Edition, Pearson Education Asia, 2011.
4. S. Lipschutz and M. Lipson, *Discrete Mathematics*, Tata McGraw Hill, 3rd Edition, 2010.
5. B. Kolman, R. C. Busby and S. C. Ross, *Discrete Mathematical structures*, 6th Edition, PHI, 2010.
6. C. L. Liu, *Elements of Discrete Mathematics*, Tata McGraw Hill, 3rd Edition, 2008.

CO-PO Articulation Matrix Discrete Mathematics Course (PCC-CSE203-T)

List of Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1. **Outline** various discrete structures and the related operations. (LOTS: Level 1: Remember)	1	-	-	-	-	-	-	-	-	-	-	-	1	1	1
CO2. **Illustrate** different discrete structures with the help of examples. (LOTS: Level 2: Understand)	1	-	-	-	-	-	-	-	-	-	-	-	1	2	1
CO3. **Apply** appropriate techniques to solve problems related to discrete structures. (LOTS: Level 3: Apply)	2	-	-	-	1	-	-	-	-	-	-	-	2	2	-
CO4. **Justify** the solutions with the help of proofs. (HOTS: Level 5: Evaluate)	3	1	-	-	2	-	-	-	-	-	-	-	3	-	-
CO5. **Combine** techniques related to discrete structures for solving real world problems. (HOTS: Level 6: Create)	3	2	-	-	2	-	-	-	1	-	-	1	3	-	-
Level of Attainments: PCC-CSE203-T															

Computer Organization and Architecture

General Course Information

Course Code: PCC-CSE204-T PCC-IT204-T Course Credits: 3 Type: Professional Core Contact Hours: 3 hours/week Mode: Lectures Examination Duration: 3 hours	**Course Assessment Methods (internal: 30; external: 70)** Two minor examinations each of 20 marks, Class Performance measured through percentage of lectures attended (4 marks) Assignment and quiz (6 marks), and end semester examination of 70 marks. For the end semester examination, nine questions are to be set by the examiner. Question number one will be compulsory and based on the entire syllabus. It will contain seven short answers type questions each of marks 2. Rest of the eight questions are to be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt any other four questions selecting one from each of the remaining four units. All questions carry equal marks.

Pre-requisites: Digital Electronics and computer systems.

About the Course:

Computer Architecture and organization describes the role of instruction set architecture in digital computer, main memory, and input/output devices. It illustrates the simple data path and control design for processors. It helps to understand the different operations and concept of instructions. It would enable the students to learn the basic function and architecture of modern computer systems.

Course Outcomes: By the end of the course students will be able to:

CO1. **outline** the general concepts of digital electronics and computer organisation and architecture. (LOTS: Level 1: Remember)
CO2. **discuss** the basic components and their interfacing.(LOTS: Level 2: Understand)
CO3. **apply** instructions for performing different operations. (LOTS: Level 3: Apply)
CO4. **analyse** the effect of addressing modes on the execution time of a program.(HOTS: Level 4: Analyse)
CO5. **contrast** different types of memory, their architecture and access methods. (HOTS: Level 5: Evaluate)
CO6. **Design** of simple computer with different instruction sets. (HOTS: Level 6: Create)

Course Content

Unit I

Basic Principles: Boolean algebra and Logic gates, Combinational logic blocks (Adders, Subtractors, Multiplexers, Encoders, decoders, demultiplexers, KMaps), Sequential logic blocks (Flip-Flops, Registers, Counters); Flynn's classification of computers (SISD, MISD, MIMD); Performance metrics: MIPS, MFLOPS; CPU Architecture types: computer register, (accumulator, register, stack, memory/ register) detailed data path of a typical register based CPU.

Unit II

Computer Organization: Store program control concept, Instruction codes, timing and control, instruction cycle; type of instructions: memory reference, register reference, I/O reference; Basics of Logic Design, accumulator logic, Control memory; Micro Programmed Control: address sequencing, micro-instruction formats, micro-program sequencer, Implementation of control unit.

Unit III

Instruction Set Architecture & Parallelism: Instruction set based classification of processors (RISC, CISC, and their comparison); Stack Organization, Instruction Formats; addressing modes: register, immediate, direct, indirect, indexed; Operations in the instruction set: Arithmetic and Logical, Data Transfer, Control Flow; Types of interrupts; Introduction to Parallelism: Goals of parallelism (Exploitation of concurrency, throughput enhancement); Amdahl's law; Instruction level parallelism (pipelining, super scaling –basic features); Processor level parallelism (Multiprocessor systems overview).

Unit IV

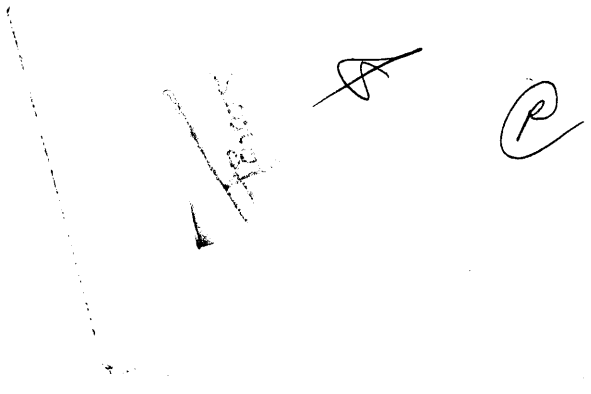
Memory Hierarchy & I/O Techniques: The need for a memory hierarchy (Locality of reference principle, Memory hierarchy in practice: Cache, main memory and secondary memory, Memory parameters: access/ cycle time, cost per bit); Main memory (Semiconductor RAM & ROM organization, memory expansion, Static & dynamic memory types); Cache memory (Associative & direct mapped cache organizations; input-output interface, mode of transfer, DMA (Direct memory transfer).

Text and Reference Books:

1. Mano, M. Morris, *Digital Logic and Computer Design*, Prentice Hall of India Pvt. Ltd., 1981.
2. M. Morris Mano, *Computer System Architecture*, Prentice Hall of India Pvt. Ltd., 1993.
3. Milles J. Murdocca, Vincent P. Heuring, *Computer Architecture and Organization, An Integrated Approach*, JohnWiley & Sons Inc., 2007.
4. William Stallings, 10th edition, *Computer Organization and Architecture*, Prentice Hall, 2016.
5. Heuring, V.P., Jordan, H.F., *Computer Systems Design and Architecture*, Addison Wesley,1997.
6. R.P Jain, *Modern Digital Electronics*, 3rd Edition , Tata McGraw Hill,, 2003.

CO-PO Articulation Matrix Computer Organization and Architecture Course (PCC-CSE204-T)

List of Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1. **outline** the general concepts of digital electronics and computer organisation and architecture. (LOTS: Level 1: Remember)	1	-	-	-	-	-	-	-	-	-	-	-	2	-	-
CO2. **discuss** the basic components and their interfacing. (LOTS: Level 2: Understand)	1	-	-	-	-	-	-	-	-	-	-	-	3	-	-
CO3. **Apply** instructions for performing different operations. (LOTS: Level 3: Apply)	2	-	-	-	-	-	-	-	-	-	-	-	3	-	-
CO4. **Analyse** the effect of addressing modes on the execution time of a program. (HOTS: Level 4: Analyse)	2	2	-	1	-	-	-	-	-	-	-	1	3	-	-
CO5. **Contrast** different types of memory, their architecture and access methods. (HOTS: Level 5: Evaluate)	2	2	-	1	-	-	-	-	-	-	-	1	3	-	-
CO6. **Design** of simple computer with different instruction sets. (HOTS: Level 6: Create)	3	2	-	-	2	-	-	-	-	-	-	-	3	-	-
Level of Attainments PCC-CSE204-T															



Data Structures and Algorithms using C/C++Lab.

General Course Information

Course Code: PCC-CSE201-P/ PCC-IT201-P Course Credits: 2 Type: Professional Core Lab. Course Contact Hours: 4 hours/week Mode: Lab practice and assignments	**Course Assessment Methods (internal: 30; external: 70)** The internal and external assessment is based on the level of participation in lab. Sessions and the timely submission of lab experiments/assignments, the quality of solutions designed for the assignments, the performance in VIVA-VOCE, the quality of lab. file and ethical practices followed. The internal examination is conducted by the course coordinator. The external examination is conducted by external examiner appointed by the Controller of Examination in association with the internal examiner appointed by the Chairperson of the Department.

Pre-requisites: Programming in C language.

About the Course:

This lab. course involves implementation of basic and advance data structures and various operations on these data structures. The objective of the lab course is to train the students to solve the problems related to data structures and choose the appropriate data structure for solving computational problem efficiently.

Course Outcomes: By the end of the lab course a student would be able to:

CO1. **Implement** various data structures and the related operations. (LOTS: Levels 3: Apply)
CO2. **Analyse** space and time complexity of algorithms. (HOTS: Level 4: Analyse)
CO3. **Compare** solutions on the basis of the appropriateness of data structure used and the efficiency of the operations implemented. (HOTS: Level 5: Evaluate)
CO4. **Integrate** knowledge of data structures to solve real world problems related to data structure and algorithms. (HOTS: Level 6: Create)
CO5. **Create** written records for the given assignments with problem definition, design of solution and conclusions. (HOTS: Level 6: Create)
CO6. **Demonstrate** ethical practices while solving problems individually or in groups (LOTS: Level 3: Apply).

List of experiments/assignments

1. Two assignments related to creating and manipulating matrices and linear lists.
2. Two assignments associated with linked list, operations on linked lists and their applications.
3. Two assignments on array and linked implementation of stacks and queues.
4. Two assignments on trees and their applications.
5. Two assignments on graphs and their applications.
6. Two assignments on different searching and sorting methods along with their complexity analysis.
7. One assignment on challenging problems on data structures to be given in groups.

Note:

The actual experiments/assignments will be designed by the course coordinator. One assignment should be designed to be done in groups of two or three students. The assignments must meet the objective of the course and the levels of the given course outcomes. The list of assignments and schedule of submission will be prepared by the course coordinator at the beginning of the semester.

CO-PO Articulation Matrix Data Structures and Algorithms Lab. Course (PCC-CSE201-P)

List of Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1. **Implement** various data structures and the related operations. (LOTS: Levels 3: Apply)	2	-	-	-	1	-	-	-	2	-	-	-	3	-	-
CO2. **Analyse** space and time complexity of algorithms. (HOTS: Level 4: Analyse)	2	2	-	-	1	-	-	-	1	-	-	-	3	-	-
CO3. **compare** solutions on the basis of the appropriateness of data structure used and the efficiency of the operations implemented. (HOTS: Level 5: Evaluate)	2	2	-	-	1	-	-	-	1	-	-	-	3	-	-
CO4. **integrate** knowledge of data structures to solve real world problems related to data structure and algorithms. (HOTS: Level 6: Create)	3	2	3	-	-	-	-	-	3	-	-	-	3	-	-
CO5. **Create** written records for the given assignments with problem definition, design of solution and conclusions. (HOTS: Level 6: Create)	-		-	-	-	-	-	-	-	3	-	-	-	-	-
CO6. **Demonstrate** ethical practices while solving problems individually or in groups (LOTS: Level 3: Apply).	-		-	-	-	-	-	3	-	-	-	3	-	-	-
Level of Attainments: PCC-CSE201-P															

Object Oriented Programming using C++ Lab.

General Course Information

Course Code: PCC-CSE202-P/ PCC-IT202-P Course Credits: 2 Type: Professional Core Lab. Course Contact Hours: 4hours/week Mode: Lab practice and assignments	**Course Assessment Methods (internal: 30; external: 70)** The internal and external assessment is based on the level of participation in lab. sessions and the timely submission of lab experiments/assignments, the quality of solutions designed for the assignments, the performance in VIVA-VOCE, the quality of lab. file and ethical practices followed. The internal examination is conducted by the course coordinator. The external examination is conducted by external examiner appointed by the Controller of Examination in association with the internal examiner appointed by the Chairperson of the Department.

Pre-requisites: Problem solving using C Lab.

About the course:

The lab course provides the opportunity to students to solve problems using Object Oriented Framework in C++ language. This includes implementing the concepts of data abstraction, data hiding, and encapsulation, reuse of code and, compile and runtime polymorphism.

Course Outcomes: By the end of the course students will be able to:

CO1. **implement** problems with object oriented framework. (LOTS: Level 3: Apply)
CO2. **analyse** the structure of programs for modular design. (HOTS: Level 4: Analyse)
CO3. **evaluate** robustness of a program by testing it on test/use cases. (HOTS: Level 5: Evaluate)
CO4. **design** class hierarchies for implementing inheritance/polymorphism. (HOTS: Level 6: Create)
CO5. **create** a lab record of assignments including problem definitions, design of solutions and conclusions. (HOTS: Level 6: Create)
CO6. **demonstrate** ethical practices and solve problems individually or in a group. (LOTS: Level 3: Apply)

List of assignments

1. Create two classes **DM** and **DB** which store the value of distances. **DM** stores distances in meters and centimeters and **DB** in feet and inches. Write a program that can read values for the class objects and add one object of **DM** with another object of **DB**. Use a friend function to carry out the addition operation. The object that stores the results maybe a **DM** object or **DB** objects, depending on the units in which the result is required. The display should be in the format of feet and inches or meters and centimeters depending on the object on display.
2. Create a class rational which represents a numerical value by two double values- NUMERATOR & DENOMINATOR. Include the following public member Functions:
   - constructor with no arguments (default).
   - constructor with two arguments.
   - void reduce () that reduces the rational number by eliminating the highest common factor between
   - the numerator and denominator.
   - Overload + operator to add two rational number.

- Overload >> operator to enable input through cin.
- Overload << operator to enable output through cout.

Write the main () function to test all the functions in the class.

3. A hospital wants to create a database regarding its indoor patients. The information to be stored includes

   a) Name of the patient b) Date of admission c) Disease d) Date of discharge

   Create a structure to store the date (year, month and day as its members). Create a base class to store the above information. The member function should include functions to enter information and display a list of all the patients in the database. Create a derived class to store the age of the patients. List the information about all the pediatric patients (less than twelve years in age).

4. Make a class **Employee** with a name and salary. Make a class **Manager** inherited from **Employee**. Add an instance variable named department of type string. Supply a method to **to String**that prints the manager's name, department and salary. Make a class **Executive** inherited from **Manager**. Supply a method **to String**that prints the string **"Executive"** followed by the information stored in the **Manager** superclass object. Supply a test program that tests these classes and methods.

5. Imagine a tollbooth with a class called 'tollBooth'. The two data items are of type unsigned int to hold the total number of cars, and a type double to hold the total amount of money collected. A constructor initializes both to 0. A member function called 'payingCar()' increments the car total and adds 0.50 to the cash total. Another function, called 'nopayCar ()', increments the car total but adds nothing to the cash total. Finally, a member function called displays the two totals. Include a program to test this class. This program should allow the user to push one key to count a paying car, and another to count a nonpaying car. Pushing the ESC kay should cause the program to print out the total cars and total cash and then exit.

6. Write a function called 'revers_it()' that reverses a string (an array of char). Use a for loop that swaps the first and last characters, then the second and next to last characters and so on. The string should be passed to 'revers_it ()' as an argument. Write a program to exercise 'revers_it ()'. The program should get a string from the usercall of revers_it () function and print out the result. Use an input method that allows embedded blanks. Test the program with phrase, "*Guru Jambheshwar University of Science & Technology, Hisar*".

7. Write a program related to file handling with all the exception handling provisions.

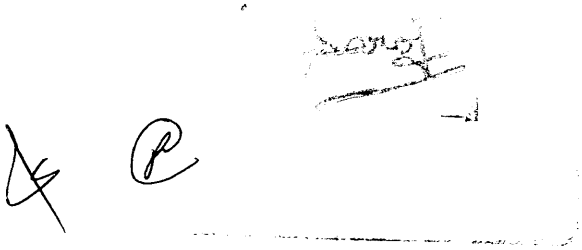
8. C++ program to write and read time in/from binary file using fstream. Use exception handling wherever possible.

9. Write a program to implement string classusing STL.

10. Write a program to implement run time polymorphism.

Note:

The experiments/assignments may vary from session to session and will be designed by the course coordinator. The assignments must meet the objective of the course and the levels of the given course outcomes. The course coordinator will provide the schedule for submission of the assignment.



CO-PO Articulation Matrix Object Oriented Programming using C++ Lab. (PCC-CSE202-P)

List of Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1. **Implement** problems with object oriented framework. (LOTS: Level 3: Apply)	2	2	-	-	1	-	-		2	-	-	2	3	-	-
CO2. **Analyse** the structure of programs for modular design. (HOTS: Level 4: Analyse)	2	2	-	-	2	-	-	-	-	-	-	-	3	-	-
CO3. **evaluate** robustness of a program by testing it on test/use cases. (HOTS: Level 5: Evaluate)	2	2	-	-	2	-	-	-	-	-	-	-	3	-	-
CO4. **Design** class hierarchies for implementing inheritance/polymorphism. (HOTS: Level 6: Create)	3	-	1	-	2	-	-	-	-	-	-	2	3	-	-
CO5. **Create** a lab record of assignments including problem definitions, design of solutions and conclusions. (HOTS: Level 6: Create)	-	-	-	-	-	-	-		-	3	-	-	-	-	-
CO6. **Demonstrate** ethical practices and solve problems individually or in a group. (LOTS: Level 3: Apply)	-	-	-	-	-	-	-	3	3	-	-	3	-	-	-
Level of Attainments: PCC-CSE202-P															

Microprocessors and Interfacing

General Course Information:

Course Code: PCC-CSE205-T/ PCC-IT205-T Course Credits: 3 Type: Professional Core Contact Hours: 3 Mode: Lecture (L) Examination Duration: 3 Hours	**Course Assessment Methods (internal: 30; external: 70)** Two minor examinations (20 marks), Class Performance measured through percentage of lectures attended (4 marks), assignments (6 marks), and the end- semester examination (70 marks). For the end semester examination, nine questions are to be set by the examiner. A candidate is required to attempt 5 questions in all. All questions carry equal marks. Question number 1 will be compulsory and based on the entire syllabus. It will contain seven parts of 2 marks each. Question numbers 2 to 9 will be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt the remaining four questions by selecting one question from each of the four units.

Pre-requisites: The students are expected to have a strong background in the Computer Organization Digital System Design.

About the Course

By the end of the course students will be able to:

CO1. **outline** the architecture of 8085 and 8086 Microprocessor. (LOTS: Level 1: Remember)
CO2. **discuss** the basic principles of addressing modes, pin diagrams. (LOTS: Level 2: Understand)
CO3. **describe** the functionality of various peripheral chip (LOTS: Level 2: Understand)
CO4. **apply** the concepts of interfacing of Memory, Input/output with Microprocessor. (LOTS: Level 3: Apply)
CO5. **compare** and **contrast** the working of 8085 and 8086 microprocessors. (HOTS: Level 5: Evaluate)
CO6. **develop** Assembly Language programs for 8085 and 8086 microprocessor.(HOTS: Level 6: Create)

Course Content

Unit-I

Introduction to microprocessor, 8085 microprocessor architecture, instruction set, interrupt structure, Architecture of 8086, block diagram of 8086, details of sub-blocks such as EU, BIU; memory segmentation and physical address computations, program relocation.

Unit II

Addressing modes, instruction formats, pin diagram and description of various signals, Instruction execution timing, assembler instruction format, data transfer instructions, arithmetic instructions, branch instructions, looping instructions, NOP and HLT instructions, flag manipulation instructions, logical instructions, shift and rotate instructions, directives and operators.

Unit III

Assembler directives, Programming with an assembler, Programming examples, coding style, the art of assembly language programming.

Software Development with Interrupts, Introduction to Stack, Stack Structure of 8086, Introduction to Subroutines, Recursion, MACROS. BIOS (Basic Input/Output System), DOS (Disk Operating System).

Unit IV

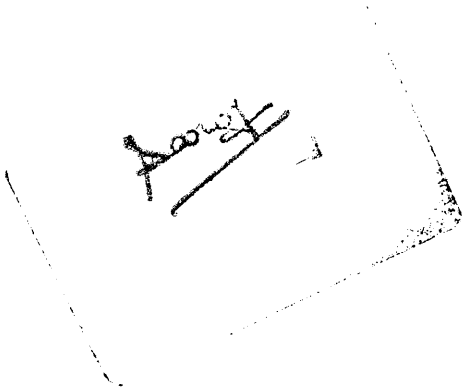
The 8255 PPI chip: Architecture, control words, modes and examples.

Introduction to DMA process, 8237 DMA controller.

8259 Programmable interrupt controller, Programmable interval timer chips.

Text and Reference Books:

1. Ramesh S Gaonkar; *Microprocessor Architecture, Programming & Applications with 8085*, Wiley Eastern Ltd., 5th edition, 2002.
2. Brey, *The Intel Microprocessors 8086- Pentium processor*, PHI, 8th edition, 2009.
3. Douglas V Hall; *Microprocessors and Interfacing*, TMH, 2000.
4. Triebel & Singh; *The 8088 & 8086 Microprocessors-Programming, interfacing, Hardware & Applications*, PHI , 4th edition, 2003.
5. Yu-Chang Liu & Glenn A Gibson ; *Microcomputer systems: the 8086/8088 Family: architecture, Programming & Design,* PHI, 1986.



CO-PO Articulation Microprocessor and Interfacing Course (PCC-CSE205-T)

List of Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1. **Outline** the architecture of 8085 and 8086 Microprocessor. (LOTS: Level 1: Remember)	1	-	-	-	-	-	-	-	-	-	-	-	3	-	-
CO2. **Discuss** the basic principles of addressing modes, pin diagrams. (LOTS: Level 2: Understand)	2	-	-	-	-	-	-	-	-	-	-	-	3	-	-
CO3. **describe** the functionality of various peripheral chip (LOTS: Level 2: Understand)	2	1	-	-	-	-	-	-	-	-	-	-	3	-	-
CO4. **apply** the concepts of interfacing of Memory, Input/output with Microprocessor. (LOTS: Level 3: Apply)	2	1	-	-	-	-	-	-	-	-	-	-	3	-	-
CO5. **compare** and **contrast** the working of 8085 and 8086 microprocessors. (HOTS: Level 5: Evaluate)	2	-	1	1	2	-	-	-	-	-	-	1	3	-	-
CO6. **develop** Assembly Language programs for 8085 and 8086 microprocessor.(HOTS: Level 6: Create)	3	2	2	-	2	-	-	-	1	-	-	1	3	-	-
Level of Attainments PCC-CSE205-T															

Computer Networks

General Course Information

Course Code: PCC-CSE206-T/ PCC-IT301-T Course Credits: 3 Type: Professional Core Contact Hours: 3 hours/week Mode: Lectures (L) Examination Duration: 3 hours	**Course Assessment Methods (internal: 30; external: 70)** Two minor examinations (20 marks), Class Performance measured through percentage of lectures attended (4 marks), assignments (6 marks), and the end- semester examination (70 marks). For the end semester examination, nine questions are to be set by the examiner. A candidate is required to attempt 5 questions in all. All questions carry equal marks. Question number 1 will be compulsory and based on the entire syllabus. It will contain seven parts of 2 marks each. Question numbers 2 to 9 will be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt the remaining four questions by selecting one question from each of the four units.

Pre-requisites: Basic knowledge of Digital and Analog Communication.

About the Course:

This course has been designed with an aim to provide students with an overview of the concepts and fundamentals of data communication and computer networks. The learner is given an opportunity to grasp various algorithms for routing of data, forwarding data and switching the data from hop to hop. Layered Architecture adds value to the subject contents.

Course Outcomes: By the end of the course students will be able to:

CO1. **outline** various models, topologies and devices of Computer Networks. (LOTS: Level 1: Remember)
CO2. **explain** the functions of various layers in Network Reference Model. (LOTS: Level 2: Understand)
CO3. **apply** different network concepts in various network communication protocols. (LOTS: Level 3: Apply)
CO4. **analyse** performance of various protocols in different scenarios. (HOTS: Level 4: Analyse)
CO5. **design** network for an organisation. (HOTS: Level 6: Create)

Course Content

Unit I

Data communication: Components, Data representation and Data flow; Network: Uses, Topologies, Network Services, OSI and TCP/IP Reference Models; Network categories: LAN, MAN, WAN; Guided Transmission Media, Wireless Transmission Media, Switching Techniques: Circuit Switching, Packet Switching, Message Switching, Networking Devices: Hubs, Repeaters, Bridges, Modems, Switches, Routers, and Gateways.

Unit II

Data Link Layer-design issues, Framing & Error Handling: Framing Protocols, Error detection and correction mechanisms; Flow Control Protocols: Stop-and-wait, Sliding Window protocols: Go-back-N and Selective Repeat; Medium Access sub layer: Channel allocation methods, Multiple Access Communication: Random Access-ALOHA, Slotted-ALOHA, CSMA, CSMA-CD, LAN Standards: Ethernet, Fast Ethernet & Gigabit Ethernet.

Unit III

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

Internetworking: IPV4 and IPV6, IP Addressing (Classful Addressing, Private IP Addresses, Classless Addressing, Sub-netting), ARP, RARP, ICMP, Internet Routing Protocol.

Unit IV

Transport Layer: Transport layer Services: Addressing, Multiplexing, Flow control, Buffering and Error control. Internet Transport Protocols: UDP, TCP, TCP Segment, TCP Connection.

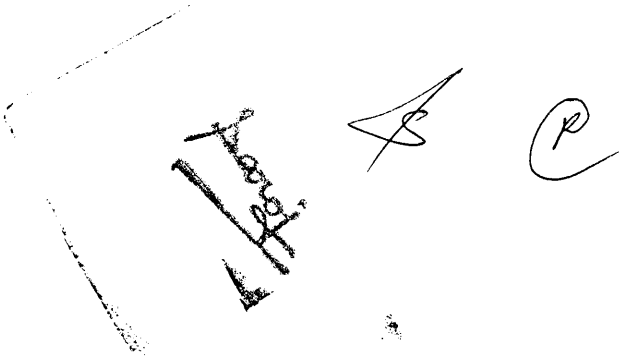
Application Layer: Introduction to DNS, FTP, TELNET, HTTP, SMTP, Electronic Mail, WWW and Multimedia.

Text and Reference Books:

1. Andrew S Tanenbaum, *Computer Networks*, 5th Edition, Pearson publications, 2010.
2. Forouzan, *Data Communication and networking* ,5th Edition, Tata McGrawHill, 2012.
3. William Stalling, *Data & Computer Communication* 6th edition, LPE Pearson Education, 2013.
4. Todd Lammle, *CCNA Study Guide*, 6th Edition, 2013.
5. RFCs and Internet Drafts available from Internet Engineering Task Force.

CO-PO Articulation Matrix Computer Networks Course (PCC-CSE206-T)

List of Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1. **outline** various models, topologies and devices of Computer Networks. (LOTS: Level 1: Remember)	1	-	-	-	-	-	-	-	-	-	-	-	-	3	-
CO2. **explain** the functions of various layers in Network Reference Model. (LOTS: Level 2: Understand)	1	-	-	-	-	-	-	-	-	-	-	-	-	3	-
CO3. **Apply** different network concepts in various network communication protocols. (LOTS: Level 3: Apply)	2	1	-	-	2	-	-	-	-	-	-	-	-	3	-
CO4. **Analyse** performance of various protocols in different scenarios. (HOTS: Level 4: Analyse)	2	2	2	1	2	-	-	-	-	-	-	-	-	3	-
CO5. **Design** network for an organisation. (HOTS: Level 6: Create)	3	2	2	-	2	-	-	-	-	-	-	-	-	3	-
Level of Attainments PCC-CSE206-T															



Database Management System

General Course Information

Course Code: PCC- CSE207-T/ PCC-IT207-T Course Credits: 3 Type: Professional Core Contact Hours: 3 hours/week Mode: Lectures (L) Exam Duration: 3 hours	**Course Assessment Methods (internal: 30; external: 70)** Two minor examinations (20 marks), Class Performance measured through percentage of lectures attended (4 marks), assignments (6 marks), and the end- semester examination (70 marks). For the end semester examination, nine questions are to be set by the examiner. A candidate is required to attempt 5 questions in all. All questions carry equal marks. Question number 1 will be compulsory and based on the entire syllabus. It will contain seven parts of 2 marks each. Question numbers 2 to 9 will be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt the remaining four questions by selecting one question from each of the four units.

Prerequisite: Knowledge of UNIX, Windows, a programming language and data structures

About the Course:

This course includes a detailed coverage of principles of database design and models. Students learn querying a database using SQL, normalization techniques, transaction processing etc.

Course Outcomes: By the end of the course students will be able to:

CO1. **describe** fundamental elements of Database Management System. (LOTS: Level 1: Remember)
CO2. **discuss** principles of relational Database modelling. (LOTS: Level 2: Understanding)
CO3. **apply** SQL for designing queries for Relational Databases. (LOTS: Level 3: Apply)
CO4. **contrast** various concurrency control and recovery techniques with concurrent transactions in DBMS. (HOTS: Level 5: Evaluate)
CO5. **design** models of databases using ER modelling and normalization for real life applications.(HOTS: Level 6: Create)

Course Content

Unit I

Overview: Overview of File Systems and Database Systems, Characteristics of the Data Base Approach, Database users, Advantages and Disadvantages of a DBMS, Responsibility of Database Administrator.

Data Base Systems Concepts and Architecture: DBMS architecture and various views of Data, Data Independence, Database languages, Data Models: Relational Database Model, Hierarchical Data Model, Network Data Model, Schemas and Instances.

Unit II

E-R Model: Entity Types, Attributes & Keys, Relationships, Roles and Structural Constraints, E-R Diagrams, Reduction of an E-R Diagram to Tables. Relational Model and Query Language: Overview of Relational Database, Key Integrity Constraints, Relational Algebra, Relational Calculus, SQL fundamentals, Basic Operators, Missing information and NULL values, Advanced SQL features

Unit III

Relational Database Design: Overview of normalization, Database Anomalies, Candidate and Super Key, Functional Dependencies, Integrity Constraints, Decomposition, Normal forms: First, Second, Third Normal, Boyce Codd, Normal Form, Multi-valued Functional Dependencies and Fourth Normal Form, Join Dependencies and Fifth Normal Form, Denormalization.

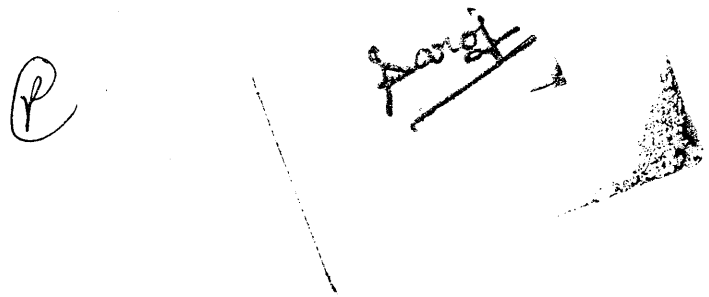
Unit - IV

Concurrency Control Techniques: Overview of database Transactions, Transaction states, ACID properties of a Transaction, Transaction Recovery, Concurrency Control, Locking Techniques, Time-stamp ordering, Multi-version Techniques, Deadlock, Recovery Techniques in centralized DBMS.

DDBMS Design: Replication and Fragmentation Techniques.

Text and Reference Books:

1. Elmasri, R., and Navathe, S. B., *Fundamentals of Database Systems*, 3rd Edition, Addison Wesley, 2002.
2. Silberschatz, A., Korth, H. F., and Sudarshan, S., Database *System Concepts,* McGraw Hill, 2011.
3. Pannerselvam R., *Database Management Systems*, 2nd Edition, PHI Learning, 2011.
4. Desai, B. C., *An Introduction to Database System*, Galgotia Publication, 2010.
5. Leon, A., and Leon, M., *Database Management Systems*, Ist Edition, Vikas Publishing, 2009.
6. Mata-Toledo, R., Cushman, P., Sahoo, D., *Database Management Systems*, Schaums' Outline series, TMH, 2007.



CO-PO Articulation Matrix Database Management System Course (PCC-CSE207-T)

List of Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1. **Describe** fundamental elements of Database Management System. (LOTS: Level 1: Remember)	1	-	-	-	-	-	-	-	-	-	-	-	3	-	-
CO2. **discuss** principles of relational Database modeling. (LOTS: Level 2: Understanding)	1	-	-	-	-	-	-	-	-	-	-	-	3	-	-
CO3. **Apply** SQL for designing queries for Relational Databases. (LOTS: Level 3: Apply)	1	-	-	-	2	-	-	-	-	-	-	-	3	-	-
CO4. **contrast** various concurrency control and recovery techniques with concurrent transactions in DBMS. (HOTS: Level 5: Evaluate)	1	2	-	-	-	-	-	-	-	-	-	-	3	-	-
CO5. **Design** models of databases using ER modelling and normalization for real life applications. (HOTS: Level 6: Create)	3	2	3	1	2	-	-	-	-	-	-	-	3	-	-
Level of Attainments PCC-CSE-207-T															

Analysis and Design of Algorithms

General Course Information

Course Code: PCC-CSE208-T/ PCC-IT208-T Course Credits: 3 Type: Professional Core Contact Hours: 3 hours/week Mode: Lectures (L) Examination Duration: 3 hours	**Course Assessment Methods (internal: 30; external: 70)** Two minor examinations (20 marks), Class Performance measured through percentage of lectures attended (4 marks), assignments (6 marks), and the end- semester examination (70 marks). For the end semester examination, nine questions are to be set by the examiner. A candidate is required to attempt 5 questions in all. All questions carry equal marks. Question number 1 will be compulsory and based on the entire syllabus. It will contain seven parts of 2 marks each. Question numbers 2 to 9 will be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt the remaining four questions by selecting one question from each of the four units.

Pre-requisites: Knowledge of Data Structure and a Programming Language

About the Course:

This Course focus on effective and efficient design of algorithms. In this course various algorithm design techniques and their analysis is to be studied. After studying this course, a student is expected to apply better techniques for solving computational problems efficiently and prove it analytically.

Course Outcomes: By the end of the course students will be able to:

CO1. **state** terminology and concepts algorithmic techniques. (LOTS: Level 1: Remember)
CO2. **discuss** various algorithmic techniques. (LOTS: Level 2: Understand)
CO3. **apply** appropriate algorithmic techniques to solve computational problems. (LOTS: Level 3: Apply)
CO4. **analysing** algorithms for their efficiency by determining their complexity. (HOTS: Level 4: Analyse)
CO5. **compare** the pros and cons of applying the different algorithmic techniques to solve problems. (HOTS: Level 5: Evaluate)
CO6. **formulate** efficient and effective algorithmic solutions for different real- world problems. (HOTS: Level: 6 Create)

Course Content

Unit I

Algorithms, Algorithms as a technology, Insertion sort, Analyzing algorithms, asymptotic notations, Divide and Conquer: General method, binary search, merge sort, quick sort, Strassen"s matrix multiplication algorithms and analysis of algorithms for these problems.

Unit II

Sorting and Data Structures: Heapsort, Hash Tables, Red and Black Trees, Greedy Method: General method, knapsack problem, minimum spanning trees, single source paths and analysis of these problems.

Unit III

Dynamic Programming: General method, matrix chain multiplication, longest common subsequence, optimal binary search trees,

Back Tracking: General method, 8 queen's problem, graph colouring, Hamiltonian cycles, Analysis of these problems.

Unit IV

Branch and Bound: Method, O/I knapsack and traveling salesperson problem, NP Completeness: Polynomial time, NP-completeness and reducibility, NP-complete problems.

Text and Reference Books:

1. Thomas H. Cormen, Charles E. Leiserson, Ronalde L. Rivest, Clifford Stein, *Introduction to Algorithms*, MIT press, 3rd Edition, 2009.
2. Ellis Horowitz, Satraj Sahni, Sanguthevar Rajasekaran, *Fundamental of Computer Algorithms*, Galgotia publication Pvt. Ltd., 1999.
3. S. Dasgupta, C. Papadimitriou, and U. Vazirani, *Algorithms*, McGraw-Hill Higher Education, 2006.

CO-PO Articulation Matrix Analysis and Design of AlgorithmsCourse (PCC-CSE208-T)

List of Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1. **state** terminology and concepts algorithmic techniques. (LOTS: Level 1: Remember)	1	-	-	-	-	-	-	-	-	-	-	-	2	-	-
CO2. **discuss** various algorithmic techniques. (LOTS: Level 2: Understand)	1	-	-	-	-	-	-	-	-	-	-	-	3	-	-
CO3. **Apply** appropriate algorithmic techniques to solve computational problems. (LOTS: Level 3: Apply)	2	-	1	-	-	-	-	-	-	-	-	-	3	2	2
CO4. **analysing** algorithms for their efficiency by determining their complexity. (HOTS: Level 4: Analyse)	3	2	1	-	2	-	-	-	-	-	-	-	3	2	2
CO5. **compare** the pros and cons of applying the different algorithmic techniques to solve problems. (HOTS: Level 5; Evaluate)	3	2	1	-	-	-	-	-	-	-	-	-	3	2	2
CO6. **formulate** efficient and effective algorithmic solutions for different real- world problems. (HOTS: Level: 6 Create)	3	3	2	2	-	-	-	-	-	-	-	-	3	2	2
Level of Attainments PCC-CSE208-T															

Software Engineering

General Course Information

Course Code: PCC-CSE209-T/ PCC-IT209-T Course Credits: 3 Type: Professional Core Contact Hours: 3hours/week Mode: Lectures (L) Examination Duration: 3 hours	**Course Assessment Methods (internal: 30; external: 70)** Two minor examinations (20 marks), Class Performance measured through percentage of lectures attended (4 marks), assignments (6 marks), and the end- semester examination (70 marks). For the end semester examination, nine questions are to be set by the examiner. A candidate is required to attempt 5 questions in all. All questions carry equal marks. Question number 1 will be compulsory and based on the entire syllabus. It will contain seven parts of 2 marks each. Question numbers 2 to 9 will be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt the remaining four questions by selecting one question from each of the four units.

Pre-requisites: Knowledge of algorithms, flow charts and a programming language.

About the Course:

Software Development is generally a quite complex and time-consuming process. Moreover, depending on the nature and complexity of the software requirements, Software Engineering plays an important role. This course will help the students to understand the systematic approach to requirement analysis, design, development, operations and maintenance of software systems. Besides this, it also guides students in developing the optimal software systems.

Course Outcomes: By the end of the course students will be able to:

CO1. **define** the various concepts related to software engineering. (LOTS: Level 1: Remember)
CO2. **demonstrate** the use of stages of various Software Life Cycle Models. (LOTS: Level 2: Understanding)
CO3. **apply** the Software Requirement Analysis and Software Design Process. (LOTS: Level 3: Apply)
CO4. **analyse** the size, cost, complexity, reliability, quality and maintenance of a software system. (HOTS: Level 4: Analyse)
CO5. **construct** software model according to the requirements of a customer. (HOTS: Level 6: Create)

Course Content

Unit I

Introduction: Software Crisis, Software Process, Evolution of Software Engineering, Software Characteristics, Software Metrics and SDLC. Software Life Cycle Models: Water Fall Model, Increment Process Model, Evolutionary Process Models, Unified Process. Selection of Life Cycle Model.

Software Requirements, Analysis and Specifications: Requirement Engineering, Requirements Elicitation, Requirements Analysis: Data Flow Diagram, Data Dictionary, Entity-Relationship Diagrams, Decision Table, Decision Tree and Structured Charts. Requirements Documentation and Requirements validation.

Unit II

Software Project Management: Size Estimation, Cost Estimation, Constructive Cost Model (COCOMO), Putnam Resource Allocation Model. Software Risk Management: Software Risks, Risk Identification, Risk Projection, Risk Refinement, Risk Mitigation, Monitoring, and Management, RMMM Plan.

Unit III

Software Design: Software Design Fundamentals, Modularity, Design Principles, Strategy of Design, Function Oriented Design, and Object Oriented Design, IEEE Recommended Practice for Software Design Descriptions.

Software Quality: Basic Concepts, ISO 9126, McCall's Quality Factors, Software Quality Assurance, SQA Activities, Software Review Process, Formal Technical Review, ISO 9000 Quality Standards, and CMM.

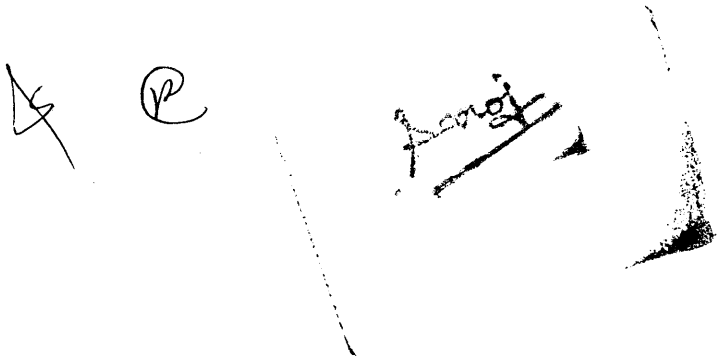
Unit IV

Software Testing: Testing fundamentals, Verification and Validation, Test Plan, Test Case, Levels of Software Testing: Unit Testing, Integration Testing, Top Down and Bottom up Testing Integration Testing, Alpha and Beta Testing, System Testing, White Box Testing and Black Box Testing, Debugging and Software Testing Tools.

Maintenance and Reengineering: Software Maintenance, Software Supportability, Reengineering, Business Process Reengineering, Software Reengineering, Reverse Engineering, Restructuring, Forward Engineering and The Economics of Reengineering.

Text and Reference Books:

1. K. K. Aggarwal and Yogesh Singh, *Software Engineering*, 3rd Edition, New Age International Publishers Ltd., Reprint 2014.
2. Roger S. Pressman, *Software Engineering: A Practitioners Approach* 7th Edition, Mc Graw Hill Education, 2014.
3. Rajib Mall, *Fundamental of Software Engineering*, Prentice Hall India, 2004.
4. Pankaj Jalote, *An integrated Approach to Software Engineering*, 3rd Edition, Narosa Publications, 2014.
5. Ian Sommerville, *Software Engineering*, 10th Edition, Addison-Wesley, 2015.
6. Carlo Ghezzi, Mehdi Jazayeri and Dino Mandrioli, *Fundamentals of Software Engineering*, 2nd Edition, Pearson, 2007.
7. Waman S Jawadekar, *Software Engineering-Principles and Practice*, Tata McGraw-Hill, 2004.



CO-PO Articulation Matrix Software Engineering Course (PCC-CSE209-T)

List of Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1. **Define** the various concepts related to software engineering.(LOTS: Level 1: Remember)	1	–	–	–	–	–	–	–	–	–	–	–	3	–	–
CO2. **Demonstrate** the use of stages of various Software Life Cycle Models. (LOTS: Level 2: Understanding)	1	–	–	–	–	–	–	–	–	–	–	–	3	–	–
CO3. **Apply** the Software Requirement Analysis and Software Design Process. (LOTS: Level 3: Apply)	2	1	1	–	2	–	–	–	–	–	2	–	3	–	–
CO4. **Analyse** the size, cost, complexity, reliability, quality and maintenance of a software system. (HOTS: Level 4: Analyse)	2	3	2	–	2	–	–	–	–	2	2	–	3	–	–
CO5. **Construct** software model according to the requirements of a customer. (HOTS: Level 6: Create)	3	3	2	–	3	–	–	–	2	2	3	–	3	–	–
Level of Attainments PCC-CSE209-T															

Java Programming

General Course Information

Course Code: PCC-CSE210-T/ PCC-IT-210-T Course Credits: 3 Type: Professional Core Contact Hours: 3 hours/week Mode: Lectures (L) Examination Duration: 3 Hours	**Course Assessment Methods (internal: 30; external: 70)** Two minor examinations (20 marks), Class Performance measured through percentage of lectures attended (4 marks), assignments (6 marks), and the end- semester examination (70 marks). For the end semester examination, nine questions are to be set by the examiner. A candidate is required to attempt 5 questions in all. All questions carry equal marks. Question number 1 will be compulsory and based on the entire syllabus. It will contain seven parts of 2 marks each. Question numbers 2 to 9 will be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt the remaining four questions by selecting one question from each of the four units.

Pre-requisites: The course assumes knowledge of Object-Oriented Concepts and programming in any Object-Oriented language.

About the Course:

Java is a general-purpose, concurrent, class-based, object-oriented computer programming language that is specifically designed to have as few implementation dependencies as possible. The aim of this course is to provide the students basic knowledge about object-oriented development and in-depth knowledge about syntax and programming techniques in Java.The course is very comprehensive and cover all the important Java concepts, e.g., Java basics, Object-Oriented Programming, Multithreading, File handling, Exception handling and more.

Course Outcomes: By the end of the course students will be able to:

CO1. **list** object oriented characteristics peculiar to JAVA programming. (LOTS: Level 1: Remember)
CO2. **describe** object-oriented principles and paradigms implemented by Java programming language. (LOTS: Level 2: Understand)
CO3. **apply** object-oriented principles for solving problems using JAVA. (LOTS: Level 3: Apply)
CO4. **identify** classes, interfaces methods, hierarchy in the classes for a given programming problem in JAVA. (HOTS: Level 4: Analyse)
CO5. **design** Graphical User Interface applications and Web based applications in Java by importing applet, AWT and SWING packages. (HOTS: Level 6: Create)

Course Content

Unit I

Object-Oriented Programming Concepts: Object, Classes, Instantiation, Reuse, Procedural and object oriented programming paradigms, Features of object-oriented programming: Encapsulation, Abstraction, Inheritance, and Polymorphism.

Java Programming Fundamentals: History of Java, Features of Java architecture, java architecture security, Garbage collections and Memory Management. Java programming language syntax, constants, variables, data types, operators, expressions .type conversion and casting. Control statements: if-else, for, while, & do-while loops and switch statements. Methods, constructors, access specifiers and modifiers, Overloading methods and Overloading constructors. Recursion, building strings, exploring string class.

Unit II

Implementing OOP : Inheritance – Inheritance hierarchies, super and sub classes, super keyword, Implementing inheritance, overriding methods, and interfaces. Implementing multiple inheritance using interfaces. Polymorphism- dynamic binding, Method Overriding, Abstract Methods and Classes.

Exception Handling: Exceptions in java, exception classes, built-in exceptions, try, catch and finally statements. Multiple catch statements, throw and throws statement. Creating and handling user-defined exceptions.

Working with Packages: Packages-Defining, Creating and Accessing a Package, Java API Packages, Using System Packages, Understanding CLASSPATH, importing packages.

Unit III

Working with GUI: Introduction to an Applet, life cycle of an applet, passing parameters to applets, applet security issues. AWT and Swing components, Layout Managers.

Event Handling: Delegation event model, event listeners and event handlers, Working with frames, working with buttons and Text Fields, working with combo-boxes, working with List Boxes etc.

Threads: Multithreading, Thread class and Runnable interface. life cycle of a thread, Thread priority, thread synchronization.

Unit IV

File Handling: File class and Random Access File class, Input and output streams, character streams, Object serialization, Serializable interface, Remote Method Invocation (RMI).

Database Connectivity: Introduction to SQL statements, ODBC/JDBC API: Connection, Statements, Prepared Statements, Record Set and execute statements.

Text and Reference Books:

1. Paul Deital, Harvey Deital, *Java™: How to Program*,9th Edition, Pearson Education (Prentice Hall), 2012.
2. Herbert Schildt, *Java™: The Complete Reference*,7th Edition, McGraw-Hill, 2007.
3. Kathy Sierra, Bert Bates, *Head First Java*, 2nd Edition, O'Reilly, 2005.
4. Ralph Bravaco , Shai Simoson, *Java Programming From the Ground Up*, Tata McGraw-Hill, 2009.
5. Sachin Malhotra, Saurabh Chaudhary, Programming in Java, Oxford University Press, 2011.
6. E. Balagurusamy, *Programming with Java: A Premier*, 3rd Edition, Tata McGraw-Hill, New Delhi, 2007.

CO-PO Articulation Matrix Java Programming Course (PCC-CSE210-T)

List of Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1. **List** object oriented characteristics peculiar to JAVA programming. (LOTS: Level 1: Remember)	1	–	–	–	–	–	–	–	–	–	–	–	3	–	–
CO2. **Describe** object-oriented principles and paradigms implemented by Java programming language. (LOTS: Level 2: Understand)	1	–	–	–	–	–	–	–	–	–	–	–	3	–	–
CO3. **Apply** object-oriented principles for solving problems using JAVA. (LOTS: Level 3: Apply)	2	2	–	–	–	–	–	–	–	–	–	–	3	–	–
CO4. **identify** classes, interfaces methods, hierarchy in the classes for a given programming problem in JAVA. (HOTS: Level 4: Analyse)	2	3	3	–	2	–	–	–	–	–	–	–	3	2	–
CO5. **Design** Graphical User Interface applications and Web based applications in Java by importing applet, AWT and SWING packages. (HOTS: Level 6: Create)	3	3	3	–	2	–	–	–	2	–	–	–	3	–	–
Level of Attainments PCC-CSE210-T															

Microprocessors and Interfacing Lab.

General Information

Course Code: PCC-CSE205-P PCC-IT205-P Course Credits: 1 Type: Professional Core Lab. Course Contact Hours: 2 hours/week Mode: Lab practice and assignments	**Course Assessment Methods (internal: 30; external: 70)** The internal assessment is based on the percentage of lab sessions attended (4 marks), timely submission of lab experiments/assignments and the quality of solutions provided in the assignments (16 marks), and an internal VIVA-VOCE (10 marks) conducted towards the end of semester. The external examination is of 70 marks. The break-up of marks for external examination is based on quality of lab reports (20 marks), quality of solution(s) for the given problem(s) at the time of examination (written work + execution of program(s)) (30) and VIVA-VOCE examination (20).

Pre-requisites: Basic concepts of Digital Electronics and Logic Design, Computer Organization

About the Course:

The objective of the lab course is to equip the students to design the Assembly Language programs for 8085 and 8086 microprocessors.

Course Outcomes: By the end of the course students will be able to:

CO1. **describe** the working of microprocessor kit/ TASM .(LOTS: Level 3: Apply)
CO2. **apply** interfacing of supporting chips with microprocessor. (LOTS: Level 3: Apply
CO3. **design** assembly language programs for the 8085 and 8086 microprocessors. (HOTS: Level 6: Create)
CO4. **analyse** the output of assembly language programs. (HOTS: Level 4: Analyse)
CO5. **create** lab records for the solutions of assignments. (HOTS: Level 6: Create)
CO6. **demonstrate** use of ethical practices, independent enquiry and team spirit. (LOTS: Level 3: Apply)

List of experiments/assignments

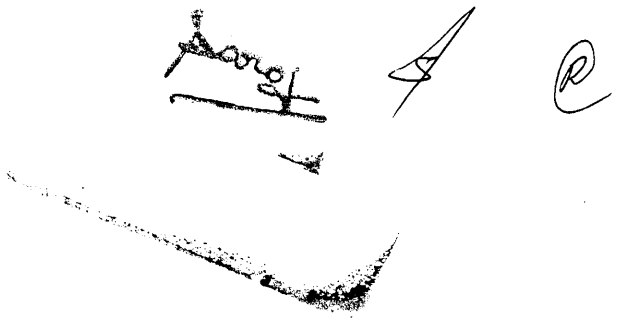
1. Two assignments to write assembly language programs using data transfer instructions
2. Two assignments to write assembly language programs using arithmetic instructions
3. Two assignments to write assembly language programs using flag manipulation instructions
4. Two assignments to write assembly language programs using shift and rotate instructions
5. Two assignments to write assembly language programs using stacks for 8086 micro-processor.
6. Two assignments to write assembly language programs using subroutines for 8086 micro-processor.
7. Two assignments on interfacing of supporting chips with 8085 and 8086 microprocessors.

Note:

The actual experiments/assignments will be designed by the course coordinator. One assignment should be designed to be done in groups of two or three students. The assignments must meet the objective of the course and the levels of the given course outcomes. The list of assignments and schedule of submission will be prepared by the course coordinator at the beginning of the semester.

CO-PO Articulation Matrix Micro-processor and Interfacing Lab. Course (PCC-CSE204-P)

List of Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1. **describe** the working of microprocessor kit/ TASM .(LOTS: Level 3: Apply)	2	-	-	-	1	-	-	-	-	-	-	1	2	-	-
CO2. **apply** interfacing of supporting chips with microprocessor. (LOTS: Level 3: Apply	2	-	-	-	1	-	-	-	-	-	-	1	3	-	-
CO3. **design** assembly language programs for the 8085 and 8086 microprocessors. (HOTS: Level 6: Create)	2	2	-	2	-	-	-	-	-	-	-	-	3	-	-
CO4. **analyse** the output of assembly language programs. (HOTS: Level 4: Analyse)	2	-	1	1	2	-	-	-	1	-	-	-	3	-	-
CO5. Create lab records for the solutions of assignments. (HOTS: Level 6: Create)	-	-	-	-	-	-	-	-	-	3	-	-	-	-	-
CO6. Demonstrate use of ethical practices, independent enquiry and team spirit. (LOTS: Level 3: Apply)	-	-	-	-	-	-	-	3	3	-	-	3	-	-	-
Level of Attainments PCC-CSE204-P															



Computer Networks Lab.

General Course Information

Course Code: PCC- CSE206-P/ PCC-IT301-P Course Credits: 1 Type: Professional Core Lab. Course Contact Hours: 2 hours/week Mode: Lab. practice and assignments	**Course Assessment Methods (internal: 30; external: 70)** The internal and external assessment is based on the level of participation in lab. sessions and the timely submission of lab experiments/assignments, the quality of solutions designed for the assignments, the performance in VIVA-VOCE, the quality of lab. file and ethical practices followed. The internal examination is conducted by the course coordinator. The external examination is conducted by external examiner appointed by the Controller of Examination in association with the internal examiner appointed by the Chairperson of the Department.

Pre-requisites: knowledge of programming, digital and analog communication.

About the Course:

This course has been designed with an aim to provide students with an overview of the concepts and fundamentals of data communication and computer networks. Students learn about various topologies, network devices, routing protocols, firewall amongst other features and devices of Computer Networks.

Course Outcomes: By the end of the course students will be able to:

CO1. **demonstrate** various network topologies and networking devices.(LOTS: Level: 3: Apply)
CO2. **justify** a particular routing protocol for any implemented data communication networks.(HOTS: Level: 5: Evaluate)
CO3. **construct** a network and implement various network protocols.(HOTS: Level: 6: Create)
CO4. **devise** solutions for various routing and switching problems in Computer Networks. (HOTS: Level: 6: Create)
CO5. **create** lab records for the solutions of the assignments. (HOTS: Level: 6: Create)
CO6. **demonstrate** ethical practices, self-learning and team spirit. (LOTS: Level: 3: Apply)

List of Experiments/assignments:

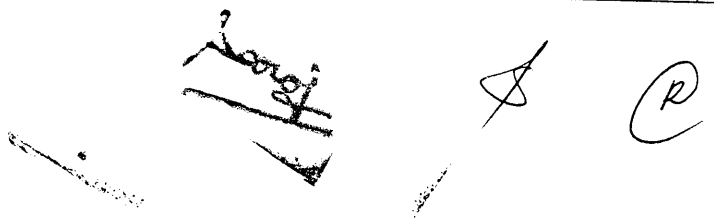
1. A) Familiarization with networking components and devices: LAN Adapters - Hubs -Switches - Routers etc.
   B)Familiarization with transmission media and Tools: Co-axial cable - UTP Cable - Crimping Tool - Connectors etc.
2. Installation and introduction of simulation tools PacketTracer/ GNS3.
3. Preparing the UTP cable for cross and direct connections using crimpingtool.
4. Introduction to various interior and exterior routing protocols.
5. Configuration of RIP protocol on routers to configure a network topology.
6. Implementation EIGRP protocol on router.
7. Implementation OSPF protocol on a larger network.
8. Configuration of ARP protocol in network.
9. Configuration of a wireless device in simulated environment.
10. Implementation BGP protocol between two different networks.
11. Implementation of static routing in simulation environment.
12. Configuration of TELNET protocol on router for remote access.
13. Configuration of access lists on network to stop unwanted traffic on network.
14. Configuration of zone based firewall in network.

Note:

The actual experiments/assignments will be designed by the course coordinator. One assignment should be designed to be done in groups of two or three students. The assignments must meet the objective of the course and the levels of the given course outcomes. The list of assignments and schedule of submission will be prepared by the course coordinator at the beginning of the semester.

CO-PO Articulation Matrix Computer Networks Lab. (PCC-CSE206-P)

List of Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO13	PSO1	PSO2	PSO3
CO1. **Demonstrate** various network topologies and networking devices. (LOTS: Level: 3: Apply)	1	1	–	–	–	–	–	–	–	–	–	–	–	3	–	–
CO2. **Justify** a particular routing protocol for any implemented data communication networks. (HOTS: Level: 5: Evaluate)	2	2	2	–	–	–	–	–	–	–	–	–	–	3	–	–
CO3. **Construct** a network and implement various network protocols. (HOTS: Level: 6: Create)	2	3	3	–	2	–	–	–	–	–	–	–	–	3	–	–
CO4. **Devise** solutions for various routing and switching problems in Computer Networks. (HOTS: Level: 6: Create)	3	3	3	3	3	–	–	–	–	–	–	–	–	3	–	–
CO5. **Create** lab records for the solutions of the assignments. (HOTS: Level: 6: Create)	–	–	–	–	–	–	–	–	–	3	--	--	–	–	–	–
CO6. **Demonstrate** ethical practices, self-learning and team spirit. (LOTS: Level: 3: Apply)	–	–	–	–	–	–	–	3	3	–	–	3	–	–	–	–
Level of Attainments PCC-CSE206-P																



Database Management System Lab.

General Course Information

Course Code: PCC-CSE-207-P/ PCC-IT207-P Course Credits: 2 Type: Professional Core Lab. Course Contact Hours: 2 hours/week Mode: Lab practice and assignments.	**Course Assessment Methods (internal: 30; external: 70)** The internal and external assessment is based on the level of participation in lab. sessions and the timely submission of lab experiments/assignments, the quality of solutions designed for the assignments, the performance in VIVA-VOCE, the quality of lab. file and ethical practices followed. The internal examination is conducted by the course coordinator. The external examination is conducted by external examiner appointed by the Controller of Examination in association with the internal examiner appointed by the Chairperson of the Department.

Pre-requisites: Exposure to a programming language, MS Access.

About the Course:

This lab. course on DBMS involves a rigorous training on Oracle programming. It provides a strong formal foundation in database concepts, technology and practice to the students to groom them into well-informed database application developers. The objective of the lab course is to develop proficiency in the execution of commands of the database design and query using Oracle.

Course Outcomes: By the end of the course students will be able to:

CO1. **implement** database problems using Oracle DML/DDL commands. (LOTS: Level 3: Apply)
CO2. **enforce** integrity constraints on a database using a state-of-the-art RDBMS. (LOTS: Level 3: Apply)
CO3. **analyse** the design of a relational database. (HOTS: Level 4: Analyse)
CO4. **design** a relational database for a given schema. (HOTS: Level 6: Create)
CO5. **create** lab assignment record that includes problem definitions, solutions, results and conclusions. (HOTS: Level 6: Create)
CO6. **demonstrate** ethical practices, self-learning and team spirit.

List of experiments/assignments:

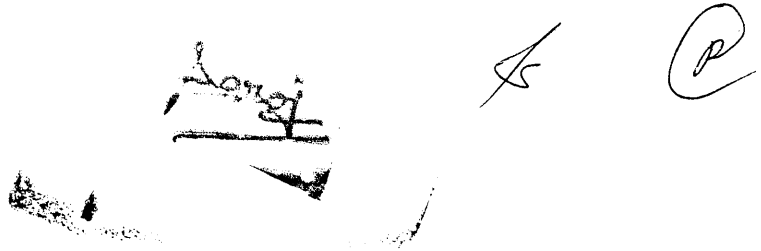
1. Use oracle software and login with valid user id and password. Explore its GUI and practice some basic commands of it.
2. Three assignments related to creation of database with tables having different fields and datatypes.
3. Two assignments on the creation of table with different types of constraints.
4. Two assignments on insert, delete and modify records from the tables.
5. Two assignments on modifying the table using the alter command.
6. Two assignments on exploring select statement using various clauses like where, order by, group by, having and aggregate functions.
7. Two assignments on the use of set operations to query the tables.
8. Two assignments on creating joins and views on the tables.
9. One assignment on generating sub-queries.

Note:

The actual experiments/assignments will be designed by the course coordinator. One assignment should be designed to be done in groups of two or three students. The assignments must meet the objective of the course and the levels of the given course outcomes. The list of assignments and schedule of submission will be prepared by the course coordinator at the beginning of the semester.

CO-PO Articulation Matrix Database Management System Lab. (PCC-CSE207-P)

List of Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1. Implement database problems using Oracle DML/DDL commands. (LOTS: Level 3: Apply)	2	1		–	2	–	–	–	–	–	–	–	3	–	–
CO2. **enforce** integrity constraints on a database using a state-of-the-art RDBMS. (LOTS: Level 3: Apply)	2	2	–	–	2	–	–	–	–	–	–	–	3	–	–
CO3. **Analyse** the design of a relational database. (HOTS: Level 4: Analyse)	3	3	1	–	2	–	–	–	–	–	–	–	3	–	–
CO4. **Design** a relational database for a given schema. (HOTS: Level 6: Create)	3	3	2	3	3	–	–	–	–	–	–	–	3	–	–
CO5. **Create** lab assignment record that includes problem definitions, solutions, results and conclusions. (HOTS: Level 6; Create)	–	–	–	–	–	–	–	–	–	3	–	–	–	–	–
CO6. Demonstrate ethical practices, self-learning and team spirit.	–	–	–	–	–	–	–	3	3	–	–	3	–	–	–
Level of Attainments PCC-CSE207-P															



Java Programming Lab.

General Course Information

Course Code: PCC-CSE210-P/ PCC-IT210-P Course Credits: 2 Type: Professional Core Lab. Course Contact Hours: 4 hours/week Mode: Lab practice and assignments	**Course Assessment Methods (internal: 30; external: 70)** The internal and external assessment is based on the level of participation in lab. sessions and the timely submission of lab experiments/assignments, the quality of solutions designed for the assignments, the performance in VIVA-VOCE, the quality of lab. file and ethical practices followed. The internal examination is conducted by the course coordinator. The external examination is conducted by external examiner appointed by the Controller of Examination in association with the internal examiner appointed by the Chairperson of the Department.

Pre-requisites: The course assumes knowledge of Object-Oriented Concepts and programming.

About the Course:

This Java course will provide a strong understanding of basic Java programming elements and data abstraction using problem representation and the object-oriented framework. The objective of the lab course is to inculcate proficiency in students to design and develop market-based software applications.

Course Outcomes: By the end of the course students will be able to:

CO1. **implement** Java programs using object oriented concepts for problem solving. (LOTS: Level 3: Apply)
CO2. **detect** syntax and logical errors in java programs (HOTS: Level 4: Analyse)
CO3. **apply** exception handling for making robust JAVA code. (HOTS: Level 3: Apply)
CO4. **design** java applications using File I/O and GUI. (HOTS: Level 6: Create)
CO5. **create** lab record of the solutions of assignments that includes problem definitions, solutions and conclusions. (HOTS: Level 6: Create)
CO6. **demonstrate** ethical practices, self-learning and team spirit. (LOTS: Level 3: Apply)

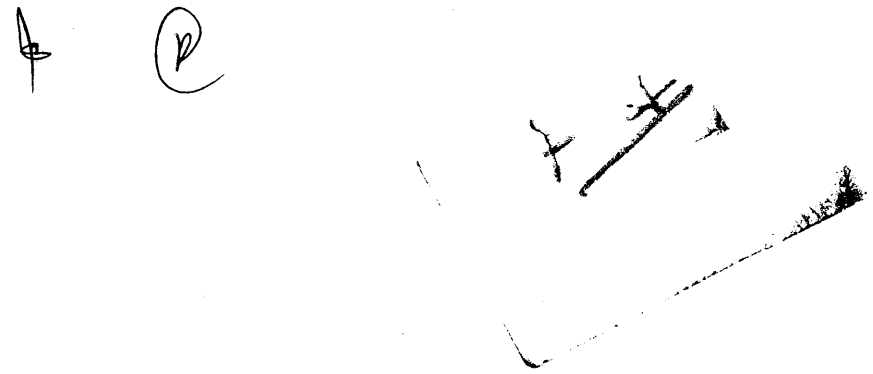
List of experiments/assignments:

1. Use eclipse or NetBeans platform and acquaint with the various menus, create a test project, add a test class and run it to see how you can use auto suggestions and auto fill functionalities. Try code formatter and code refactoring like renaming variables, methods and classes. Try debug step by step with a small program of about 10 to 15 lines which contains at least one if else condition and a for loop.
2. Two assignments illustrating class, objects, methods, arrays and various data types in java.
3. Two assignments on the use of control, looping statements and user defined functions.
4. One assignment illustrating the implementation of various forms of inheritance.
5. One assignment on method overloading.
6. One assignment on polymorphism and method overriding.
7. One assignment on implementing exception handling.
8. One assignment to illustrate interfaces in java.
9. One assignment to create package in java.

10. One assignment to design of multithreaded programs in java.
11. One new assignment on event handling.
12. Two assignments related to java applets.
13. One assignment to design a GUI application.
14. One assignment to access and update data from a database using JDBC.

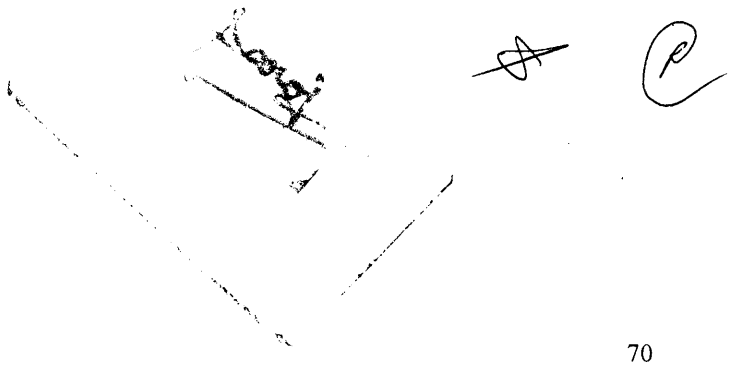
Note:

The actual experiments/assignments will be designed by the course coordinator. One assignment should be designed to be done in groups of two or three students. The assignments must meet the objective of the course and the levels of the given course outcomes. The list of assignments and schedule of submission will be prepared by the course coordinator at the beginning of the semester.



CO-PO Articulation Matrix Java Programming Lab. (PCC-CSE210-P)

List of Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1. **Implement** Java programs using object oriented concepts for problem solving. (LOTS: Level 3: Apply)	2	2	2	1	2	–	–	–	–	–	–	–	3	–	–
CO2. **Detect** syntax and logical errors in java programs (HOTS: Level 4: Analyse)	–	–	2	1	2	–	–	–	–	–	–	–	3	–	–
CO3. **Apply** exception handling for making robust JAVA code. (HOTS: Level 3: Apply)	2	2	1	1	–	–	–	–	–	–	–	–	3	–	–
CO4. **Design** java applications using File I/O and GUI. (HOTS: Level 6: Create)	3	3	3	1	3	–	–	–	–	3	–	–	3	–	–
CO5. **Create** lab record of the solutions of assignments that includes problem definitions, solutions and conclusions. (HOTS: Level 6: Create)	–	–	–	–	–	–	–	3	3	–	–	3	–	–	–
CO6. **Demonstrate** ethical practices, self-learning and team spirit. (LOTS: Level 3: Apply)	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Level of Attainments PCC-CSE210-P															



Computer Graphics

General Course Information

Course Code: PCC-CSE301-T/ PEC-IT409-T Course Credits: 3 Type: Professional Core Contact Hours: 3 hours/week Mode: Lectures (L) Examination Duration:3 hours	**Course Assessment Methods (internal: 30; external: 70)** Two minor examinations (20 marks), Class Performance measured through percentage of lectures attended (4 marks), assignments (6 marks), and the end- semester examination (70 marks). For the end semester examination, nine questions are to be set by the examiner. A candidate is required to attempt 5 questions in all. All questions carry equal marks. Question number 1 will be compulsory and based on the entire syllabus. It will contain seven parts of 2 marks each. Question numbers 2 to 9 will be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt the remaining four questions by selecting one question from each of the four units.

Pre-requisites: Programming skills in C/C++ and Data Structures.

About the Course:

This course involves studying graphic techniques, algorithms and imaging models. Moreover, students learn about the techniques for clipping, cropping, representing 2-D and 3-D objects.

Course Outcomes: By the end of the course students will be able to:

CO1. **state** basic concepts related to graphics. (LOTS: Level 1: Remember)
CO2. **describe** the principles of creating graphical objects and graphical user interface applications. (LOTS: Level 2: Understand)
CO3. **apply** 2-D and 3-D transformations (rotation, scaling, translation, shearing) on geometric objects. (LOTS: Level 3: Apply)
CO4. **use** different techniques for clipping and filling geometric objects. (LOTS: Level 3: Apply)
CO5. **compare** different graphics algorithms for different geometric objects. (HOTS: Level 4: Analyse)
CO6. **create** user-friendly interfaces for computer applications. (HOTS: Level 6: Create)

Course Content

Unit I

Introduction to Computer Graphics: What is Computer Graphics, Computer Graphics Applications, Computer Graphics Hardware and software, Two dimensional Graphics Primitives: Points and Lines, Line drawing algorithms: DDA, Bresenham's Circle drawing algorithms: Using polar coordinates, Bresenham's circle drawing, mid-point circle drawing algorithm; Filled area algorithms: Scan-line: Polygon filling algorithm, boundary filled algorithm.

Unit II

Two/Three Dimensional Viewing: The 2-D viewing pipeline, windows, viewports, window to view port mapping; Clipping: point, clipping line (algorithms): 4 bit code algorithm, Sutherland-cohen algorithm, parametric line clipping algorithm (Cyrus Beck). Polygon clipping algorithm: Sutherland-Hodgeman polygon clipping algorithm.

Two dimensional transformations: transformations, translation, scaling, rotation, reflection, composite transformation.

Three dimensional transformations: Three-dimensional graphics concept, Matrix representation of 3-D Transformations, Composition of 3-D transformation.

Unit III

Viewing in 3D: Projections, types of projections, the mathematics of planner geometric projections, coordinate systems.

Hidden surface removal: Introduction to hidden surface removal, Z- buffer algorithm, scanline algorithm, area sub-division algorithm.

Unit IV

Representing Curves and Surfaces: Parametric representation of curves: Bezier curves, B-Spline curves. Parametric representation of surfaces; Interpolation method.

Illumination, shading, image manipulation: Illumination models, shading models for polygons, shadows, transparency. What is an image? Filtering, image processing, geometric transformation of images.

Text and reference books:

1. James D. Foley, Andeies van Dam, Stevan K. Feiner and Johb F. Hughes, *Computer Graphics Principles and Practices*, second edition, Addision Wesley, 2000.
2. Pradeep K Bhatia, *Computer Graphics*, 3rd edition, I K International Pub, New Delhi, 2013.
3. Donald Hearn and M. Pauline Baker, *Computer Graphics* 2nd Edition, PHI, 1999.
4. David F. Rogers, *Procedural Elements for Computer Graphics* Second Edition, T.M.H, 2001.
5. Alan Watt, *Fundamentals of 3Dimensional Computer Graphics*, Addision Wesley, 1999.
6. Corrign John, *Computer Graphics: Secrets and Solutions*, BPB, 1994.
7. Pilania & Mahendra, *Graphics, GUI, Games & Multimedia Projects in C*, Standard Pub., 2002.
8. N. Krishanmurthy, *Introduction to Computer Graphics,* T.M.H, 2002.

CO-PO Articulation Matrix Computer Graphics Course (PCC-CSE301-T)

List of Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1. **State** basic concepts related to graphics. (LOTS: Level 1: Remember)	1	–	–	–	–	–	–	–	–	–	–	–	2	–	–
CO2. **Describe** the principles of creating graphical objects and graphical user interface applications. (LOTS: Level 2: Understand)	1	–	–	–	–	–	–	–	–	–	–	–	3	–	–
CO3. **Apply** 2-D and 3-D transformations (rotation, scaling, translation, shearing) on geometric objects. (LOTS: Level 3: Apply)	2	1	–	–	2	–	–	–	–	–	–	–	3	–	–
CO4. **Use** different techniques for clipping and filling geometric objects. (LOTS: Level 3: Apply)	2	1	–	–	2	–	–	–	–	–	–	–	3	–	–
CO5. **Compare** different graphics algorithms for different geometric objects. (HOTS: Level 4: Analyse)	-	2	2	2	1	–	–	–	–	–	–	–	3	–	–
CO6. **Create** user-friendly interfaces for computer applications. (HOTS: Level 6: Create)	1	2	2	–	3	–	–	–	–	–	-	–	3	–	–
Level of Attainments PCC-CSE301-T			–	–	–	–	–	–	–	–	–	–	–	–	–

Python Programming

General Course Information

Course Code: PCC-CSE302-T/ PCC-IT308-T Course Credits: 3 Type: Professional Core Contact Hours: 3 hours/week Mode: Lectures (L) Examination Duration: 3 hours	**Course Assessment Methods (internal: 30; external: 70)** Two minor examinations (20 marks), Class Performance measured through percentage of lectures attended (4 marks), assignments (6 marks), and the end- semester examination (70 marks). For the end semester examination, nine questions are to be set by the examiner. A candidate is required to attempt 5 questions in all. All questions carry equal marks. Question number 1 will be compulsory and based on the entire syllabus. It will contain seven parts of 2 marks each. Question numbers 2 to 9 will be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt the remaining four questions by selecting one question from each of the four units.

Pre-requisite: Exposure to programming languages

About the Course:

Python is a popular open source programming language used for both standalone programs and scripting applications in a wide variety of domains. It is free, portable, and powerful and is both relatively easy and remarkably fun to use. In today's era Python has found great applicability in machine learning, data analytics and many other data science application. This is introductory course and covers most of the basic concepts required for basic python programming. Some of the contents are advanced may be useful for data analytics purpose.

Course Outcomes: By the end of the course students will be able to:

CO1. **outline** various basic programming constructs including operators, character sets, basic data types and control statements. (LOTS: level 1: Understand)
CO2. **explain** Python packages and their functionalities for data analysis. (LOTS: level 2: Understand)
CO3. **solve** problems using python programming. (LOTS: level 3: Apply)
CO4. **analyse** the results of data analysis or machine learning programs (HOTS: level 4: Analyse)
CO5. **evaluate** solutions according to the problem definition. (HOTS: level 5: Evaluate)
CO6. **develop** database applications in Python. (HOTS: level 6: Create)

Course Content

Unit I

Introduction to Python, History of Python, Features of Python, Python Identifiers, Python Character Set, Keywords and Indentation, Comments, Command Line Arguments, Assignment Operator, Operators and Expressions, *print()* Function, *input()* Function, *eval()* Function, Python Data Types: *int, float, complex*, Variables, Mutable vs Immutable variables, Namespaces, Decision Statements: Boolean Type, Boolean Operators, *if* statement, *else* statement, Nested Conditionals Statements, Multi-way Decision Statements (*elif* statement).

Unit II

Loop Control Statements: *While* loop, *range()* Function, *For* Loop, Nested Loops, Infinite Loop, *Break* Statement, *Continue* Statement, *Pass* Statement, Introduction to Strings, String Operations: Indexing and Slicing, Lists: Operations on List: Slicing, Inbuilt Functions for Lists, List Processing: Searching and Sorting, Dictionaries: Need of Dictionary, Operations on Directories: Creation, Addition, Retrieving Values, Deletion; Tuples, operations on Tuples, Inbuilt Functions for Tuples, Introduction to Sets, operations on sets.

Python Functions, Inbuilt functions, *Main* function, User Defined functions, Defining and Calling Function, Parameter Passing, Actual and Formal Parameters, Default Parameters, Global and Local Variables, Recursion, Passing Functions as Data, *Lamda* Function, Modules, Importing Own Module, Packages.

Unit III

Operations on File: Reading text files, read functions, *read()*, *readline()* and *readlines()*, writing Text Files, write functions, *write()* and *writelines()*,Manipulating file pointer using seek, Appending to Files.

Python Object Oriented: Overview of OOP, Classes and objects, Accessing attributes, Built-In Class Attributes, Methods, Class and Instance Variables, Destroying Objects, Polymorphism, Overlapping and Overloading of Operators, Class Inheritance: *super()*, Method Overriding, Exception Handling, *Try-except-else* clause, Python Standard Exceptions, User-Defined Exceptions

Unit IV

Databases in Python: Create Database Connection, *create, insert, read, update* and *delete* Operation, DML and DDL Operation with Databases.

Python for Data Analysis: *numpy*: Creating arrays, Using arrays and Scalars, Indexing Arrays, Array Transposition, Universal Array Function, Array Processing, Array Input and Output

Pandas: Series, Data Frame, Panel, Index objects, Re-indexing, Iteration, Sorting. *Matplotlib*: Python for Data Visualization, Visualization Section, *Sklearn*: loading of dataset, learning and predicting, Model Persistence.

Text and Reference Books:

1. Ashok Namdev Kamthane, *Programming and Problem Solving with Python*, Mc Graw Hill Education Publication, 2018.
2. John Guttag, *Introduction to Computation and Programming using Python*, Springer, Revised and Expanded version (Referred by MIT), 2013.
3. Lutz, M., *Learning Python: Powerful Object-Oriented Programming*. O'Reilly Media, Inc., 2013.
4. Michael T Goodrich and Robertto. Thamassia, Micheal S Goldwasser, *Data Structures and Algorithms in Python*, Wiley, 2016.
5. Y. Daniel Liang, *Introduction to Programming Using Python*, Pearson, 2013.
6. Reema Thareja, *Python Programming Using Problem Solving Approach* , Oxford Publications, 2017.
7. Dr. R. Nageswara Rao, Allen B. Downey, *Core Python Programming* , *Think Python*, O'Reilly Media, 2012.
8. Kenneth A. Lambert, *The Fundamentals of Python: First Programs*, Cengage Learning, 2011.

CO-PO Articulation Matrix Python Programming Course (PCC-CSE302-T)

List of Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1. **Outline** various basic programming constructs including operators, character sets, basic data types and control statements. (LOTS: level 1: Understand)	1	–	–	–	–	–	–	–	–	–	–	–	–	–	2
CO2. **Explain** Python packages and their functionalities for data analysis. (LOTS: level 2: Understand)	1	–	–	–	–	–	–	–	–	–	–	–	–	–	3
CO3. **Solve** problems using python programming. (LOTS: level 3: Apply)	3	2		2	–	3	–	–	–	–	–	–	–	–	3
CO4. **Analyse** the results of data analysis or machine learning programs (HOTS: level 4: Analyse)	2	3	2	2	–	–	–	–	–	–	–	–	–	–	3
CO5. **Evaluate** solutions according to the problem definition. (HOTS: level 5: Evaluate)	2	3	2	2	–	–	–	–	–	–	–	–	–	–	3
CO6. **Develop** database applications in Python. (HOTS: level 6: Create)	3	3	2	3		3	–	–	–	–	–	–	3		3
Level of Attainments PCC-CSE302-T															

High Speed Network Technologies

General Course Information

Course Code: PCC-CSE303-T/ PEC-IT305-T Course Credits: 3 Type: Professional Core Contact Hours: 3 hours /week Mode: Lecture(L) Examination Duration: 3 hours	**Course Assessment Methods (internal: 30; external: 70)** Two minor examinations (20 marks), Class Performance measured through percentage of lectures attended (4 marks), assignments (6 marks), and the end- semester examination (70 marks). For the end semester examination, nine questions are to be set by the examiner. A candidate is required to attempt 5 questions in all. All questions carry equal marks. Question number 1 will be compulsory and based on the entire syllabus. It will contain seven parts of 2 marks each. Question numbers 2 to 9 will be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt the remaining four questions by selecting one question from each of the four units.

Pre-requisites: Basic knowledge of computer networks, layers of OSI reference model, protocols at different layers of OSI reference model.

About the course:

High Speed Network Technologies is a professional core course based around Network Architectures, protocols used across the layers, techniques used in communication and modes of data transfer. The course deals with creating High Speed Networks for any organization/institute with its various phases/life cycles.

Course Outcomes: By the end of the course students will be able to:

CO1. **define** different high speed network technologies. (LOTS: Level 1: Remember)
CO2. **explain** working of different wired / wireless technologies suitable for LAN and WAN communication. (LOTS: Level 2: Understand)
CO3. **illustrate** the mapping of OSI reference model to different high speed technologies and Internet Suite of Protocols. (LOTS: Level 3: Apply)
CO4. **analyze** the performance of different high speed technologies in different scenarios / situations. (HOTS: Level 4: Analyse)
CO5. **design** a network for any organization using high speed technologies along with Internet connectivity. (HOTS: Level 6: Create)

Course Content

Unit I (High Speed LAN)

Gigabit Ethernet: Overview of fast Ethernet, Gigabit Ethernet – overview, specifications, layered protocol architecture, frame format, network design using Gigabit Ethernet, applications, 10GB Ethernet – overview, layered protocol architecture, frame format.

Fiber Channel: Fiber channel – overview, topologies, ports, layered protocol architecture, frame structure, class of service.

UNIT II (High Speed WAN)

Frame Relay: Protocol architecture and frame format.

ISDN & B-ISDN: Channels, interfaces, addressing, protocol architecture, services.

ATM: Virtual circuits, cell switching, reference model, traffic management.

Unit III (Wireless LAN)

Wireless Networks: Existing and emerging standards, Wireless LAN (802.11), Broadband Wireless (802.16), Bluetooth (802.15) their layered protocol architecture and security. Mobile Networks – GSM, CDMA.

Unit IV (Internet Suite of Protocols)

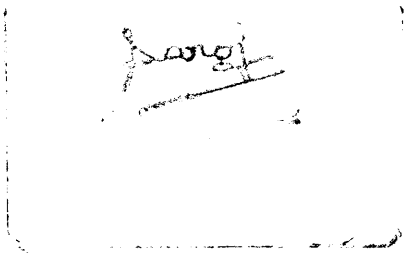
Internet Layer: IPV4 and IPV6, IP addressing, IP classes, CIDR.

Transport Layer: UDP/TCP protocols & architecture, TCP connection management.

Application Layer: DNS, E-Mail, Voice over IP.

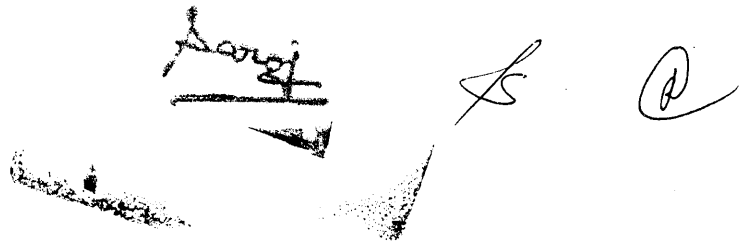
Text and Reference Books:

1. Jochen Schiller, *Mobile Communication*, 2nd Edition, Pearson,2009.
2. Andrew S Tanenbaum, *Computer Networks*, 5th Edition, Pearson 2013.
3. William C Y Lee, *Mobile Communication Engineering: Theory and Applications*, 2nd Edition, McGraw Hill, 1997.



CO-PO Articulation Matrix High Speed Network Technologies Course (PCC-CSE303-T)

List of Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1. **Define** different high speed network technologies. (LOTS: Level 1: Remember)	1	–	–	–	–	–	–	–	–	–	–	–		2	
CO2. **Explain** working of different wired / wireless technologies suitable for LAN and WAN communication. (LOTS: Level 2: Understand)	1	–	–	–	–	–	–	–	–	–	–	–		3	
CO3. **Illustrate** the mapping of OSI reference model to different high speed technologies and Internet Suite of Protocols. (LOTS: Level 3: Apply)		2	–	–	–	–	–	–	–	–	–	–		3	
CO4. **Analyze** the performance of different high speed technologies in different scenarios / situations. (HOTS: Level 4: Analyse)	2	2	2	2	2	–	–	–	–	–	–	–		3	
CO5. **Design** a network for any organization using high speed technologies along with Internet connectivity. (HOTS: Level 6: Create)	2	3	3	2	3	–	–	–	2	–	–	2		3	
Level of Attainments PCC-CSE303-T															



Cryptography and Network Security

General Course Information

Course Code: PCC-CSE304-T/ PEC-IT402-T Course Credits: 3 Type: Professional Core Contact Hours: 3hours/week Mode: Lectures (L) Examination Duration: 3 hours	**Course Assessment Methods (internal: 30; external: 70)** Two minor examinations (20 marks), Class Performance measured through percentage of lectures attended (4 marks), assignments (6 marks), and the end- semester examination (70 marks). For the end semester examination, nine questions are to be set by the examiner. A candidate is required to attempt 5 questions in all. All questions carry equal marks. Question number 1 will be compulsory and based on the entire syllabus. It will contain seven parts of 2 marks each. Question numbers 2 to 9 will be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt the remaining four questions by selecting one question from each of the four units.

Pre-requisites: Basic knowledge of Number systems, Complexity Theory, Computer Networks.

About the Course:

The increase in techniques to penetrate into systems has led to variety of information and Network attacks, To mitigate the exploitation of the vulnerabilities leading to these attacks we need to adopt robust security architecture into our premises. We have to choose between various security technologies such as cryptography, Digital Signatures, Key Management, Program Security, Database security, Wifi security. In the current scenario we require to secure end-to-end devices, Networks, Networking devices and clouds.

Course outcomes: By the end of the course students will be able to:

CO1. **recognize** need of cryptography and cryptographic Algorithms.(LOTS: Level 1: Remember)
CO2. **represent** security in terms of various techniques and algorithms. (LOTS: Level2: Understand)
CO3. **apply** mathematical techniques to cryptography for solving problems related to security issue. (LOTS: Level 3: Apply)
CO4. **identify** various types of attacks for their mitigation/proactive and reactive treatment. (HOTS: Level 4: Analyze)
CO5. **judge** the security of an organization/institute by means of Network security devices/models/controls. (HOTS: Level 5: Evaluate)
CO6. **integrate** different types of securities under one environment and evaluate its performance.(HOTS: Level 6: Create)

Course Content

Unit I

Cryptography: Overview of classical cryptosystems, stream and block ciphers, ciphers & cipher modes, Substitution Ciphers: Mono-alphabetic Substitution and Poly-alphabetic Substitution, Transposition Ciphers: Rail Fence, ScyTale, Book cipher, Vernam cipher, Vigenere Tabluae, Hill Cipher. Cryptanalysis of Classical Cryptosystems.

Unit II

Mathematical Foundations: Elementary Number theory, Finite fields, Groups and Subgroups, Matrix representations, Symmetric matrices and diagonalization, Number theory: Divisibility, gcd, prime numbers, primality testing. Congruences, solution of congruences, Chinese remainder theorem, Fermat and Euler's theorem, Modular Arithmetic and its properties, Modular exponentiation.

Unit III

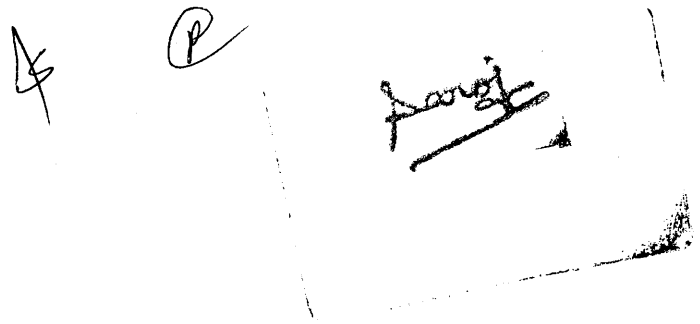
Cryptographic Algorithms and techniques: Private/Symmetric Key cryptography: DES and its variants, AES, Feistel networks, Modes of operation, Public/Asymmetric Key Cryptography: RSA Algorithm, Elliptic Curve Cryptography. Diffie Hellman Key Exchange Algorithm, Digital Signatures, Knapsack Algorithm, Public Key Infrastructure, Kerberos, secret sharing schemes, Digital Certificates, X.509 Certificates.

Unit IV

Network Security: Attacks: types, detection, mitigation. Network Security Foundations, Defense Models, Access Control: Authentication and Authorization Controls, Network Architecture, Network Device Security, Wireless Security, Firewalls, Intrusion Detection Systems, Network Role-Based Security: Email- PGP, PEM, S-MIME. Proxy servers. SSL, TLS. SET, SHTTP, IPSec. Virtual Private Networks security.

Text and Reference Books

1. William Stallings, *Cryptography and Network security-Principles and Practices*, Pearson Education, Ninth Indian Reprint 2005.
2. Charlie Kaufman , *Network Security : Private communication in Public World*, Prentice-Hall International, Inc. April 2008.
3. Roberta Bragg, Mark Rhodes-Ousley, Keith Strassberg, *The Complete Reference Network Security*, McGraw hill Education, 2004.
4. Charles P. Fleeger, *Security in Computing*, 2nd Edition, Prentice Hall International Inc., 1996.



CO-PO Articulation Matrix Cryptography and Network Security (PCC-CSE304-T)

List of Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1. **Recognize** need of cryptography and cryptographic Algorithms. (LOTS:. Level 1: Remember)	1	–	–	–	–	–	–	–	–	–	–	–	–	3	–
CO2. **Represent** security in terms of various techniques and algorithms. (LOTS: Level2: Understand)	1	–	–	–	–	–	–	–	–	–	–	–	–	3	–
CO3. **Apply** mathematical techniques to cryptography for solving problems related to security issue. (LOTS: Level 3: Apply)	3	3	2	2	2	–	–	–	–	–	–	–	–	3	–
CO4. **Identify** various types of attacks for their mitigation/proactive and reactive treatment. (HOTS: Level 4: Analyze)	2	2	2	3	3	–	–	–	–	–	–	–	–	3	–
CO5. **Judge** the security of an organization/institute by means of Network security devices/models/controls. (HOTS: Level 5: Evaluate)	3	3	2	3	3	–	–	3	–	3	–	–	–	3	–
CO6. **Integrate** different types of securities under one environment and evaluate its performance. (HOTS: Level 6: Create)	3	3	2	3	3									3	–
Level of Attainments PCC-CSE304-T															

Computer Graphics Lab.

General Course Information

Course Code: PCC-CSE301-P/ PEC-IT409-P Course Credits: 1 Type: Professional Core Lab. Course Mode: Lab practice and assignments Contact Hours: 2 hours / week	**Course Assessment Methods (internal: 30; external: 70)** The internal and external assessment is based on the level of participation in lab. sessions and the timely submission of lab experiments/assignments, the quality of solutions designed for the assignments, the performance in VIVA-VOCE, the quality of lab. file and ethical practices followed. The internal examination is conducted by the course coordinator. The external examination is conducted by external examiner appointed by the Controller of Examination in association with the internal examiner appointed by the Chairperson of the Department.

Pre-requisites: Knowledge of C/C++ and Data Structures.

About the Course:

This lab course provides opportunity to students to implement various algorithms to do graphics. This include drawing lines, circles and ellipses. In addition, students learn to rotate, move and transform graphical objects.

Course Outcomes: By the end of the course students will be able to:

CO1. **implement** various graphics algorithms for drawing and filling of geometric objects. (LOTS: Level 3: Apply)
CO2. **demonstrate** transformation of geometric objects. (LOTS: Level 3: Apply)
CO3. **compare** strengths and weakness of various graphics algorithms. (LOTS: Level 4: Analyse)
CO4. **design** algorithms for creating scenes like flying a kite and solar eclipse. (HOTS: Level 6: Create)
CO5. **create** lab assignment record that includes problem definitions, solutions and conclusions. (HOTS: Level: 6: Create)
CO6. **demonstrate** use of ethical practices, self-learning and team spirit. (LOTS: Level 3: Apply)

List of experiments/assignments

1. A program to draw a line using Digital Differential Analyzer (DDA) Algorithm
2. A program to draw a line using Bresenham's Line Algorithm (BLA) for lines with slopes

(a) negative and less than 1.
(b) positive and less than 1.
(c) positive and greater than 1.
(d) negative and greater than 1.

3. A program to draw a circle using Bresenham's Circle Algorithm.
4. A program to draw a circle using MidPoint Circle Algorithm
5. A program to draw an ellipse using MidPoint Ellipse Algorithm.
6. A program to fill different types of geometric shapes using Flood Fill. Algorithm
7. A program to fill different types of geometric shapes using Boundary Fill Algo.
8. A program to demonstrate window to view-port mapping.
9. A program to clip a line segment using 4-bit code algorithm.

10. A program to draw a C-Curve of nth order.
11. A program that shows a scene of flying kite.
12. A program to rotate a line about its mid-point.
13. A program that shows a scene of eclipse .
14. A program that translate and rotate a circle along a horizontal line.
15. A program to rotate an ellipse about its major axis and minor axis alternatively.

Note:

The actual experiments/assignments may vary and will be designed by the course coordinator. One assignment should be designed to be done in groups of two or three students. The assignments must meet the objective of the course and the levels of the given course outcomes. The list of assignments and schedule of submission will be prepared by the course coordinator at the beginning of the semester.

CO-PO Articulation Matrix Computer Graphics Lab. Course (PCC-CSE301-P)

List of Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1. **Implement** various graphics algorithms for drawing and filling of geometric objects. (LOTS: Level 3: Apply)	2	-	3	-	2	–	–	–	–	–	–	–	3	–	–
CO2. **Demonstrate** transformation of geometric objects. (LOTS: Level 3: Apply)	2	-	-	-	2	–	–	–	–	–	–	–	3	–	–
CO3. **Compare** strengths and weakness of various graphics algorithms. (LOTS: Level 4: Analyse)	2	2	-	-	-	–	–	–	–	–	–	–	3	–	–
CO4. **Design** algorithms for creating scenes like flying a kite and solar eclipse. (HOTS: Level 6: Create)	2	2	-	2	-	–	–	–	–	–	–	–	3	–	–
CO5. **Create** lab assignment record that includes problem definitions, solutions and conclusions. (HOTS: Level: 6: Create)	–	–	–	–	–	–	–	–	–	3	–	–	–	–	–
CO6. **Demonstrate** use of ethical practices, self-learning and team spirit. (LOTS: Level 3: Apply)	–	–	–	–	–	–	–	3	3	–	–	3	–	–	–
Level of Attainments PCC-CSE301-P															

Python Programming Lab.

General Course Information

Course Code: PCC-CSE302-P/ PCC-IT308-P Course Credits: 1.5 Type: Professional Core Lab. Course Contact Hours: 3 hours/week Mode: Lab practice and assignments	**Course Assessment Methods (internal: 30; external: 70)** The internal and external assessment is based on the level of participation in lab. sessions and the timely submission of lab experiments/assignments, the quality of solutions designed for the assignments, the performance in VIVA-VOCE, the quality of lab. file and ethical practices followed. The internal examination is conducted by the course coordinator. The external examination is conducted by external examiner appointed by the Controller of Examination in association with the internal examiner appointed by the Chairperson of the Department.

Pre-requisites: Basic programming skills

About the Course:

Python is a scripting programming language known for both its simplicity and wide breadth of applications. For this reason it is considered one of the best languages for beginners. Used for everything from web development to scientific computing Python is referred to as a general purpose language by the greater programming community. The major objective of Python language is to make the students solve real word problem efficiently using python library.

Course Outcomes: By the end of the course students will be able to:

CO1. **implement** solutions to the given assignments in Python. (LOTS: Level 3: Apply)
CO2. **use** various Python packages for solving different programming problems. (LOTS: Level 3: Apply)
CO3. **devise** solutions for complex problems of data analysis and machine learning. (HOTS: Level 6: Create)
CO4. **Evaluate** the output of data analysis and machine learning models. (HOTS: Level 5: Evaluate)
CO5. **create** lab records of the solutions for the given assignments. (HOTS: Level 6: Create)
CO6. **demonstrate** use of ethical practices, self-learning and team spirit.. (LOTS: Level 3: Apply)

List of experiments/assignments

1. Install Python and explore various popular IDE like IDLE, PyCharm, and Anaconda.
2. Assignments to perform various number operations like
   a. Find maximum from a list of numbers
   b. GCD of two number
   c. Square root of a number
   d. Check number is prime or not.
   e. Print first N prime numbers
   f. Remove duplicate numbers from list
   g. Print the Fibonacci series.
3. Assignments to perform various operations on Strings like creation, deletion, concatenation.
4. Create a List L = [10, 20, 30]. Write programs to perform following operations:
   a. Insert new numbers to list L.
   b. Delete numbers from list L.
   c. Sum all numbers in list L.

d. Sum all prime numbers in list L.
e. Delete the list L.

5. Create a Dictionary D= {'Name': 'Allen', 'Age': 27, 5:123456}. Write programs to perform following operations:
   a. Insert new entry in D.
   b. Delete an entry from D.
   c. Check whether a key present in D.
   d. Update the value of a key.
   e. Clear dictionary D.
6. Two assignments on Sets to perform various operation like union, intersection, difference etc.
7. Two assignments related to searching operation like linear search, binary search.
8. Three assignments related to sorting like selection sort, bubble sort, insertion sort.
9. Demonstrate the use of dictionary for measuring student marks in five subjects and you have to find the student having maximum and minimum average marks.
10. Two assignment on usage of different available packages like random package to perform
    a. Print N random numbers ranging from 100 to 500.
    b. Print 10 random strings whose length between 3 and 5.
11. Two assignments on usage of package such as Numpy, Pandas.
12. Implement and demonstrate the functions of a simple calculator.
13. One assignment on implementing object oriented concept such as classes, inheritance, and polymorphism.
14. One assignment on file handling that how data is read and written to a file.

Note:

The actual experiments/assignments will be designed by the course coordinator. One assignment should be designed to be done in groups of two or three students. The assignments must meet the objective of the course and the levels of the given course outcomes. The list of assignments and schedule of submission will be prepared by the course coordinator at the beginning of the semester.

CO-PO Articulation Matrix Python Programming Lab. Course (PCC-CSE302-P)

List of Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1. **Implement** solutions to the given assignments in Python. (LOTS: Level 3: Apply)	2	1	–	–	3	–	–	–	–	–	–	–	–	–	3
CO2. **Use** various Python packages for solving different programming problems. (LOTS: Level 3: Apply)	2	3	–	3	3	–	–	–	–	–	–	–	–	–	3
CO3. **Devise** solutions for complex problems of data analysis and machine learning. (HOTS: Level 6: Create)	3	3	1	3	3	–	–	–	–	–	–	–	–	–	3
CO4. **Evaluate** the output of data analysis and machine learning models. (HOTS: Level 5: Evaluate)	3	3		3	3	–	–	–	–	–	–	–	–	–	3
CO5. **Create** lab records of the solutions for the given assignments. (HOTS: Level 6: Create)	–	–	–	–	–	–	–	–	–	3	–	–	–	–	–
CO6. **Demonstrate** use of ethical practices, self-learning and team spirit. (LOTS: Level 3: Apply)	–	–	–	–	–	–	–	3	3	–	–	3	–	–	–
Level of Attainments PCC-CSE302-P															

Industrial Training/Internship

General Course Information

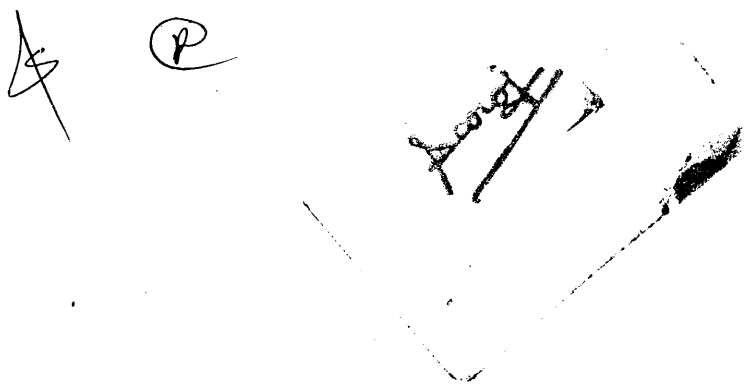
Course Code: INT-CSE301 Course Credits: 1 Mode: Industrial Training / Internship	**Course Assessment Methods (100 Marks)** An internal evaluation is done by a faculty member appointed by the Chairperson of the Department. The break-up marks for various evaluation criteria is given below. 1. Significance and originality of the problem addressed and the solution provided: 20 2. Knowledge of the problem domain and tool used (VIVA-VOCE):25 3. Report Writing: 20 4. Judgement of the skill learnt and system developed: 20 5. Level of ethics followed: 15

About the Industrial training:

Students do an Industrial Training of 4 to 6 weeks after fourth semester. They are expected to learn novel skills and develop some software application during the training period.

After doing training students will be able to:

CO1. **review** the existing systems for their strengths and weaknesses. (HOTS: Level 4: Analyse)
CO2. **address** novel problems in an original and innovative manner (HOTS: Level 6: Create)
CO3. **select and apply** modern engineering tools. (LOTS: Level 3: Apply)
CO4. **evaluate** the system developed critically with respect to the requirement analysis and other similar systems. (HOTS: Level 5: Evaluate)
CO5. **prepare** training report by organising ideas in an effective manner.
CO6. **follow** ethical practices while doing the training and writing report. (LOTS: Level 3: Apply)



CO-PO Articulation Matrix Industrial Training/Internship (INT-CSE301)

List of Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1. **Address** novel problems in an original manner using latest skills (HOTS: Level 6: Create)	3	3	3	2		1	_	_	2	_	1	_	_	_	_
CO2. **Select and apply** modern engineering tools. (LOTS: Level 3: Apply)	2	_	_	_	3	_	_	_	3	_	_	_	_	_	_
CO3. **Prepare** training report by organising ideas in an effective manner.	_	_	_	_	_	_	_	_	_	_	_	_	_	_	_
CO4. **Engage** in lifelong learning. (HOTS: Level 6: Create)	_	_	_	_	_	_	_	_	_	_	_	3	_	_	_
CO5. **Apply** ethical practices while doing the training and writing report. (LOTS: Level 3: Apply	_	_	_	_	_	_	_	3	_	3	_	_	_	_	_
Level of Attainments INT-CSE301															

Operating Systems

General Course Information

Course Code: PCC-CSE305-T/ PCC-IT206-T Course Credits: 3 Type: Professional Core Contact Hours: 3 hours/week Mode: Lectures (L) Examination Duration: 3 hours	**Course Assessment Methods (internal: 30; external: 70)** Two minor examinations (20 marks), Class Performance measured through percentage of lectures attended (4 marks), assignments (6 marks), and the end- semester examination (70 marks). For the end semester examination, nine questions are to be set by the examiner. A candidate is required to attempt 5 questions in all. All questions carry equal marks. Question number 1 will be compulsory and based on the entire syllabus. It will contain seven parts of 2 marks each. Question numbers 2 to 9 will be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt the remaining four questions by selecting one question from each of the four units.

Pre-requisites: programming in C and knowledge of computer fundamentals.

About the Course:

The objective of this course is to help students become familiar with the fundamental concepts of operating systems and provide them with enough understanding of operating system design.

Course Outcomes: By the end of the course students will be able to:

CO1. **list** various functions and design characteristics of operating systems (LOTS: Level 1: Remember)
CO2. **explain** fundamental concepts of operating systems. (LOTS: Level 2: Understand)
CO3. **apply** operating system design concepts for solving problems regarding scheduling, memory management, disk management and deadlocks etc. (LOTS: Level 3: Apply)
CO4. **analyze** the issues related to various operating systems. (HOTS: Level 4: Analyse)
CO5. **design** solutions for the memory and process management problems. (HOTS: Level 6: Create)

Course Content

Unit I

Introductory Concepts: Operating systems functions and characteristics, operating system services and systems calls, system programs, operating system structure, operating systems generations, Types of Operating systems: Batch operating system, Time-sharing OS, Distributed operating system, Realtime systems.

File Systems: Types of Files and their access methods, File allocation methods, Directory Systems: Structured Organizations, directory and file protection mechanisms, disk scheduling and its associated algorithms.

Unit II

Processes: Process concept, Process Control Block, Operations on processes, cooperating processes. CPU scheduling: Levels of Scheduling, scheduling criteria, Comparative study of scheduling algorithms, Algorithm evaluation, multiple processor scheduling. Critical-section problem, Semaphores.

Unit III

Storage Management: Storage allocation methods: Single contiguous allocation, non-contiguous memory allocation, Paging and Segmentation techniques, segmentation with paging, Virtual memory concepts, Demand Paging, Page replacement Algorithms, Thrashing.

Unit IV

Deadlock: System model, Deadlock characterization, Methods for handling deadlocks, Deadlock prevention, Deadlock avoidance, Deadlock detection, Recovery from deadlock

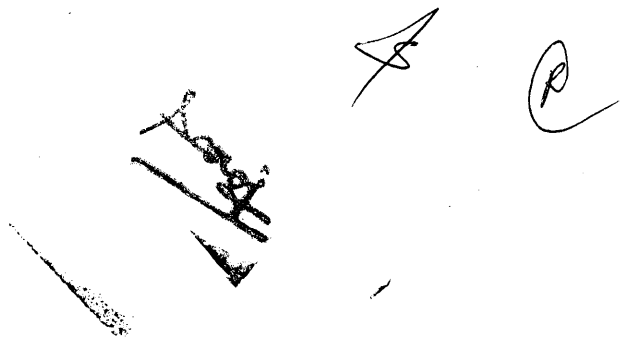
Case Studies: Comparative study of WINDOW, UNIX & LINUX system.

Text and Reference Books:

1. Silberschatz, Peter B. Galvin and Greg Gagne, *Operating System Concepts*, 8th Edition, WileyIndian Edition, 2010.
2. Andrew S Tanenbaum, *Modern Operating Systems,* Third Edition, Prentice Hall India, 2008.
3. Naresh Chauhan, *Principles of Operating Systems*, Oxford Press, 2014.
4. D.M. Dhamdhere, *Operating Systems*, 2nd edition, Tata McGraw Hill, 2010.
5. William Stallings, *Operating Systems– Internals and Design Principles*, 5th Edition, Prentice Hall India, 2000.

CO-PO Articulation Matrix Operating System Course (PCC-CSE305-T)

List of Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO6. **List** various functions and design characteristics of operating systems (LOTS: Level 1: Remember)	1	–	–	–	–	–	–	–	–	–	–	–	3	–	–
CO7. **Explain** fundamental concepts of operating systems. (LOTS: Level 2: Understand)	1	–	–	–	–	–	–	–	–	–	–	–	3	–	–
CO8. **Apply** operating system design concepts for solving problems regarding scheduling, memory management, disk management and deadlocks etc. (LOTS: Level 3: Apply)	3	1	–	–	2	–	–	–	–	–	–	1	3	–	–
CO9. **Analyze** the issues related to various operating systems. (HOTS: Level 4: Analyse)	3	2	3		2	–	–	–	–	–	–	1	3	–	–
CO10. **Design** solutions for the memory and process management problems. (HOTS: Level 6: Create)	3	2	3	2	2	–	–	–	–	–	–	–	3	–	–
Level of Attainments PCC-CSE305-T															



Formal Language and Automata Theory

General Course Information

Course Code: PCC-CSE306-T/ PCC-IT303-T Course Credits: 3 Type: Professional Core Contact Hours: 3 hours/week Mode: Lectures (L) Examination Duration: 3 hours	**Course Assessment Methods (internal: 30; external: 70)** Two minor examinations (20 marks), Class Performance measured through percentage of lectures attended (4 marks), assignments (6 marks), and the end- semester examination (70 marks). For the end semester examination, nine questions are to be set by the examiner. A candidate is required to attempt 5 questions in all. All questions carry equal marks. Question number 1 will be compulsory and based on the entire syllabus. It will contain seven parts of 2 marks each. Question numbers 2 to 9 will be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt the remaining four questions by selecting one question from each of the four units.

Pre-requisites: The students are expected to have a strong background in the fundamentals of discrete mathematics like in the areas of symbolic logic, set, induction, number theory, summation, series, combinatorics, graph, recursion, basic proof techniques.

About the Course:

Formal Languages and Automata theory presents the theoretical aspects of computer science, which lay the foundation for students of Computer Science. The course introduces some fundamental concepts in automata theory and formal languages including grammar, finite automaton, regular expression, formal language, pushdown automaton, and Turing machine.

Course Outcomes: By the end of the course students will be able to:

CO1. **define** terminology related to theory of computation. (LOTS: Level 1: Remember)
CO2. **explain** the basic concepts and applications of Theory of Computation. (LOTS: Level 2: Understand)
CO3. **apply** the principles of Theory of Computation to solve computational problems.(LOTS: Level 3: Apply)
CO4. **compare and contrast** the hierarchy of grammars (HOTS: Level 5: Evaluate).
CO5. **design** various types of automata for given problems. (HOTS: Level 6: Create)

Course Content

Unit I

Finite Automata and Regular Expressions: Finite State Systems, Basic Definitions Non-Deterministic finite automata (NDFA), Deterministic finite automata (DFA), Equivalence of DFA and NDFA Finite automata with E-moves, Regular Expressions, Equivalence of finite automata and Regular Expressions, Regular expression conversion and vice versa, Conversion of NFA to DFA by Arden's Method.

Unit II

Introduction to Machines: Concept of basic Machine, Properties and limitations of FSM. Moore and mealy Machines, Equivalence of Moore and Mealy machines.

Properties of Regular Sets: The Pumping Lemma for Regular Sets, Applications of the pumping lemma, Closure properties of regular sets, Myhill-Nerode Theorem and minimization of finite Automata, Minimization Algorithm.

Unit III

Grammars: Definition, Context free and Context sensitive grammar, Ambiguity regular grammar, Reduced forms, Removal of useless Symbols and unit production, Chomsky Normal Form (CNF), Griebach Normal Form (GNF).

Pushdown Automata: Introduction to Pushdown Machines, Application of Pushdown Machines

Unit IV

Turing Machines: Deterministic and Non-Deterministic Turing Machines, Design of T.M, Halting problem of T.M., PCP Problem.

Chomsky Hierarchies: Chomsky hierarchies of grammars, Unrestricted grammars, Context sensitive languages, Relation between languages of classes.

Computability: Basic concepts, Primitive Recursive Functions.

Text and Reference Books:

1. Hopcroaft & O. D. Ullman, R Mothwani, *Introduction to automata theory, language & computations*, AW,2001.
2. K. L. P.Mishra & N. Chandrasekaran, *Theory of Computer Sc.(Automata, Languages and computation)*, PHI, 2000.
3. Peter Linz, *Introduction to formal Languages & Automata*, Narosa, Publication, 2001.
4. Ramond Greenlaw and H. James Hoover, *Fundamentals of the Theory of Computation- Principles and Practice*, Harcourt India Pvt. Ltd., 1998.
5. H. R. Lewis & C. H. Papaditriou, *Elements of theory of Computation*, PHC, 1998.
6. John C. Martin, *Introduction to Languages and the Theory of Computation*, T.M.H., 2003.

CO-PO Articulation Matrix Formal Language and Automata Theory Course (PCC-CSE306-T)

List of Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO11. **Define** terminology related to theory of computation. (LOTS: Level 1: Remember)	1	–	–	–	–	–	–	–	–	–	–	–	3	–	–
CO12. **Explain** the basic concepts and applications of Theory of Computation. (LOTS: Level 2: Understand)	1	–	–	–	–	–	–	–	–	–	–	–	3	–	–
CO13. **Apply** the principles of Theory of Computation to solve computational problems. (LOTS: Level 3: Apply)	2	1	2		2	–	–	–	–	–	–	–	3	–	–
CO14. **Compare and contrast** the hierarchy of grammers (HOTS: Level 5: Evaluate).	3	2	2	2	2	–	–	–	–	–	–	–	3	–	–
CO15. **Design** various types of automata for given problems. (HOTS: Level 6: Create)	3	2	2	2	2	–	–	–	–	–	–	–	3	–	–
Level of Attainments PCC-CSE306-T															

Data Analytics using R

General Course Information

Course Code: PCC-CSE307-T/ PEC-IT407-T Course Credits: 2 Type: Professional Core Contact Hours: 2 hours/week Mode: Lectures (L) Examination Duration: 3 hours	**Course Assessment Methods (internal: 30; external: 70)** Two minor examinations (20 marks), Class Performance measured through percentage of lectures attended (4 marks), assignments (6 marks), and the end- semester examination (70 marks). For the end semester examination, nine questions are to be set by the examiner. A candidate is required to attempt 5 questions in all. All questions carry equal marks. Question number 1 will be compulsory and based on the entire syllabus. It will contain seven parts of 2 marks each. Question numbers 2 to 9 will be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt the remaining four questions by selecting one question from each of the four units.

Pre-requisites: Basic programming skills, Probability and Statistics

About the Course:

Data analytics is a growing and stimulating field that turns data into valuable insights. This course includes programming in R for acquiring, cleaning, visualizing and analyzing data. In addition, it also involves predictive modeling. This course will introduce students to the basic principles, tools and the craft for devising solutions for problems that come in the domain of data science. The emphasis of the course is on integration and synthesis of concepts and their applications for effective engineering solutions.

Course Outcomes: By the end of the course students will be able to:

CO1. **outline** concepts related to R programming and data analysis. (LOTS: Level 1: Remember)
CO2. **explain** the basic concepts and tools that are used to solve problems in data analytics. (LOTS: Level 2: Understand)
CO3. **interpreting** results of descriptive and inferential statistics. (LOTS: Level 2: Understand)
CO4. **apply** R programming for reading, cleaning, visualizing and analysing data. (LOTS: Level 3: Apply)
CO5. **analyse** the trends in data through exploratory data analysis. (HOTS: Level 4: Analyse)
CO6. **devise** solutions for descriptive and predictive modelling. (HOTS: Level 6: Create)

Course Content

Unit I

Introduction to R programming: Data types or objects in R, Creating and manipulating objects like factors, vectors and matrices, lists and data frames, Subsetting matrices and data frames, Vectorized operations for vectors and matrices and data frames.

Unit II

Control structure in R: If-else statements, for and while loops, loop functions like lapply, apply, sapply and mapply etc.; writing user defined functions in R. Getting data in and out of R.

Unit III

Doing basic descriptive statistics: Data types for data analysis and their mapping to R objects, Mean, Median, Mode, Quantiles, Five-point summary, Variance, Correlation and Covariance, normal distribution, uniform distribution using R, Hypothesis testing: Chi-Square test and student's T test.

Unit IV

Exploratory Data Analysis: Visualizing data through various plots and charts (bar charts, histogram, frequency polygon, scatter plot, box plots etc.), Applying KNN and Bayesian predictive models.

Text and Reference Books:

1. Hadley Wickham and Garrett Grolemund., *R for Data Science Import, Tidy, Transform and model Data*, O'Reilly, 2017.
2. Roger D. Peng, *R Programming for Data Science*, Lean Publishing, 2015.
3. Paul Teeter, *R Cookbook*, O'Reilly, 2011.
4. W. N. Venables, D. M. Smith and the R core Team, *An introduction to R, Notes on R: A Programming Environment for Data Analysis and Graphics*, version 3.3.2, 2016.
5. Michael J. Crawley, *Statistics, An introduction using R*, Second edition, John Wiley, 2015
6. Han, J., Kamber, M, Pei, J., *Data Mining Concepts and Techniques*, Third edition, Morgan Kaufmann, 2012.
7. Trevor Hastie, Robert Tibshirani, Jerome Friedman, *The Elements of Statistical Learning: Data Mining, Inference and Prediction*, Springer, 2nd edition, 2009.

CO-PO Articulation Matrix Data Analytics using R Course (PCC-CSE307-T)

List of Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1. **Outline** concepts related to R programming and data analysis. (LOTS: Level 1: Remember)	1	-	-	-	-	–	–	–	–	–	–	–	–	–	2
CO2. **Explain** the basic concepts and tools that are used to solve problems in data analytics. (LOTS: Level 2: Understand)	1	-	-	-	-	–	–	–	–	–	–	–	–	–	3
CO3. **Interpreting** results of descriptive and inferential statistics. (LOTS: Level 2: Understand)	3	2	-	3	-	–	–	–	–	–	–	–	–	–	3
CO4. **Apply** R programming for reading, cleaning, visualizing and analysing data. (LOTS: Level 3: Apply)	3	-	-	-	3	–	–	–	–	–	–	–	–	–	3
CO5. **Analyse** the trends in data through exploratory data analysis. (HOTS: Level 4: Analyse)	2	3	2	2	3	–	–	–	1	–	–	–	–	–	3
CO6. **Devise** solutions for descriptive and predictive modelling. (HOTS: Level 6: Create)	3	3	2	3	3	–	–	–	2	–	–	–	–	–	–
Level of Attainments PCC-CSE307-T															

.NET Using C#

General Course Information:

Course Code: PCC-CSE308-T/ PCC-IT302-T Course Credits: 2 Type: Professional Core Contact Hours: 2 hours/week Mode: Lectures Examination Duration: 3 hours	Course Assessment Methods (internal: 30; external: 70) Two minor examinations (20 marks), Class Performance measured through percentage of lectures attended (4 marks), assignments (6 marks), and the end- semester examination (70 marks). For the end semester examination, nine questions are to be set by the examiner. A candidate is required to attempt 5 questions in all. All questions carry equal marks. Question number 1 will be compulsory and based on the entire syllabus. It will contain seven parts of 2 marks each. Question numbers 2 to 9 will be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt the remaining four questions by selecting one question from each of the four units.

Pre-requisites: Object oriented languages

About the Course:

.NET using C# is a core and an essential advanced course for every graduate in Computer Science and Engineering. This course introduces .NET Framework basics with object oriented technology like CLS, CTS, CLR, Assembly, TypeInfo, Delegates and Reflector etc., and various operations to be implemented surrounding these features for solving real world problems. It includes ADO.NET framework with its classes for database connectivity.

Course Outeomes: By the end of the course students will be able to:

CO1. define the concepts related to .NET Framework. (LOTS: Level 1: Remember)

CO2. explain various C# constructs. (LOTS: Level 1: Understand)

CO3. apply .NET framework using C# for solving moderate/complex problems. (LOTS: Level 3: Apply)

CO4. use advanced features of C# like Reflector, and Assembly. (LOTS: Level 3: Apply)

CO5. identify logical errors in given .Net using C# programs. (LOTS: Level 3: Analyse)

CO6. Design stand-alone applications in the .NET framework using C#. (HOTS: Level 6: Create)

Course Content

Unit - I

.NET Framework: Beginning of NET Technology, Overview of .NET Framework, .NET Framework Class Libraries, NET Programming Languages, NET Namespaces and Type. Architecture of .NET Framework. Common Language Runtime (CLR) – Common Type Specification (CTS), Common Language Specifications (CLS), Assemblies of .NET Base Classes, CLR Debugger.

Unit - II

Evolution of C#: Overview of C#, C# and .NET, Similarities & Differences from JAVA, Structure of C# program. Data Types including Out and Ref, Identifiers, Variables & Constants, Flow Control and Iteration, Object-Oriented Programming in C# - Encapsulation, Inheritance, and Polymorphism, Object and Classes, Basics of C# Classes,

Unit-III

Creating DLL files, Assemblies of multiple versions. GAC Utility and Strong Name, Arrays and Strings, Boxing and Unboxing, – Exception Handling in C#, Garbage Collection & Its Stages, Files and Streams, Delegates and their usefulness and Events, Attributes, I/O in C# and Windows Applications.

Unit - IV

Architecture of ADO.NET, Database Connection, Connected and Disconnected Environment, Create Connection using ADO.NET Object Model, Connection Class, Command Class, Data Adapter Class, Dataset Class.

Text and Reference Books:

1. Benjamin Perkins, Jacob Vibe Hammer and Jon D. Reid, *C# 6 Programming with Visual studio*, Wrox publication, 2016.
2. Matt Telles, *C# Programming*, Black Book, Coriolis Group, 2001.
3. Stephen C. Perry. Atul Kahate, *Essential of .NET and Related Technologies*, Pearson Education 2009.

CO-PO Articulation Matrix .NET using C# Course (PCC-CSE308-T)

List of Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
Define the concepts related to .NET Framework. (LOTS: Level 1: Remember)	1	–	–	–	–	–	–	–	–	–	–	–	2	–	–
Explain various C# constructs. (LOTS: Level 1: Understand)	1	–	–	–	–	–	–	–	–	–	–	–	3	–	–
Apply .NET framework using C# for solving moderate/complex problems. (LOTS: Level 3: Apply)	2	1	–	–	–	–	–	–	–	–	–	–	3	–	–
Use advanced features of C# like Reflector, and Assembly. (LOTS: Level 3: Apply)	2	2	1		2	–	–	–	–	–	–	–	3	–	–
Identify logical errors in given .Net using C# programs. (LOTS: Level 3: Analyse)	2	2	1		2	–	–	–	–	–	–	–	3	–	–
Design stand-alone applications in the .NET framework using C#. (HOTS: Level 6: Create)	3	3	2	2	3	–	–	–	–	–	–	–	3	–	–
Level of Attainments PCC-CSE-308-T														–	–

Embedded System Design

General Course Information

Course Code: PEC-CSE301-T/ PEC-IT301-T Course Credits: 3 Type: Professional Elective Contact Hours: 3 Mode: Lectures (L) Examination Duration: 3 hours.	**Course Assessment Methods (internal: 30; external: 70)** Two minor examinations (20 marks), Class Performance measured through percentage of lectures attended (4 marks), assignments (6 marks), and the end- semester examination (70 marks). For the end semester examination, nine questions are to be set by the examiner. A candidate is required to attempt 5 questions in all. All questions carry equal marks. Question number 1 will be compulsory and based on the entire syllabus. It will contain seven parts of 2 marks each. Question numbers 2 to 9 will be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt the remaining four questions by selecting one question from each of the four units.

Pre-requisites: Introduction to Microprocessors and Operating Systems.

About the Course:

An embedded system is a self-contained unit that have a dedicated purpose within a device. We come across a variety of applications of embedded systems in navigation tools, telecom applications, and networking equipment to name just a few. An Embedded System's Architecture begins with a view of embedded development and how it differs from the other systems. Students learn about setting up a development environment and then move on to the core system architectural concepts, exploring pragmatic designs, boot-up mechanisms, and memory management. They are also explored to programming interface and device drivers to establish communication via TCP/IP and take measures to increase the security of IoT solutions.

Course Outcomes: By the end of the course students will be able to:

CO1. **state** the concepts related to embedded system design. (LOTS: Level 1: Remember)
CO2. **discuss** the principles of embedded systems and their applications. (LOTS: Level 2: Understand)
CO3. **apply** the principles of embedded design for problem solving. (LOTS: Level 3: Apply)
CO4. **analyze** architectural design patterns and engineering tradeoffs. (HOTS: Level 4: Analyse)
CO5. **design** architectural patterns for connected and distributed devices in the IoT. (HOTS: Level 6: Create)

Course Content

Unit I

Embedded Systems: A Pragmatic Approach- Domain definitions, Embedded Linux systems, Low-end 8-bit microcontrollers, Hardware architecture, Understanding the challenge, Multithreading, RAM, Flash memory, Interfaces and peripherals, Asynchronous UART-based serial communication:-SPI - I2C - USB, Connected systems, The reference platform, ARM reference design, The Cortex-M microprocessor

Work Environment and Workflow Optimization: Workflow overview, C compiler, Linker, Build automation, Debugger, Embedded workflow, The GCC toolchain, The cross-compiler, Compiling the compiler, Linking the executable, Binary format conversion, Interacting with the target, The GDB session, Validation, Functional tests, Hardware tools, Testing off-target, Emulators.

Unit II

Architectural Patterns: Configuration management, Revision control, Tracking activities, Code reviews, Continuous integration, Source code organization, Hardware abstraction, Middleware Application code, The life cycle of an embedded project, Defining project steps, Prototyping Refactoring, API and documentation,

The Boot-Up Procedure: The interrupt vector table, Startup code, Reset handler, Allocating the stack, Fault handlers, Memory layout, Building and running the boot code, The makefile, Running the application, Multiple boot stages, Bootloader, Building the image, Debugging a multi-stage system, Shared libraries

Unit III

Distributed Systems and IoT Architecture: Network interfaces, Media Access Control, Ethernet, Wi-Fi, Low-Rate Wireless Personal Area Networks (LR-WPAN), LR-WPAN industrial link-layer extensions, 6LoWPAN, Bluetooth, Mobile networks, Low-power Wide Area Networks (LPWANs), Selecting the appropriate network interfaces, The Internet Protocols, TCP/IP implementations, Network device drivers, Running the TCP/IP stack, Socket communication, Mesh networks and dynamic routing, Transport Layer Security, Securing socket communication, Application protocols, Message protocols, REST architectural pattern, Distributed systems; single points of failure, Summary

Unit IV

Low-Power Optimizations: System configuration, Hardware design, Clock management, Voltage control, Low-power operating modes, Deep-sleep configuration, Stop mode, Standby mode, Wake-up intervals, Measuring power, Development boards, Designing low-power embedded applications, Replacing busy loops with sleep mode, Deep sleep during longer inactivity periods, Choosing the clock speed, Power state transitions

Embedded Operating Systems: Real-time application platforms, FreeRTOS, ChibiOS, Low-power IoT systems, Contiki OS, Riot OS, POSIX-compliant systems, NuttX, Frosted, The future of safe embedded systems, Process isolation; Tock, Summary.

Text and Reference Books:

1. Daniele Lacamera, *Embedded Systems Architecture*, Packt Publishing, May 2018, ISBN: 9781788832502.
2. Raj Kamal, *Embedded Systems*, TMH, 2004.
3. M.A. Mazidi and J. G. Mazidi, *The 8051 Microcontroller and Embedded Systems*, PHI, 2004.
4. David E. Simon, *An Embedded Software Primer*, Pearson Education, 1999.
5. K.J. Ayala, , *The 8051 Microcontroller*, Penram International, 1991.
6. Rajiv Kapadia, *8051 Microcontroller & Embedded Systems*, Jaico Press, 2004.
7. Prasad, *Embedded Real Time System*, Wiley Dreamtech, 2004.
8. John B. Peatman, *Design with PIC Microcontrollers*, Pearson Education Asia, 2002.
9. Wayne Wolf, *Computers as components: Principles of Embedded Computing System Design*, Morgan Kaufman Publication, 2000.
10. Tim Wilmshurst, *The Design of Small-Scale embedded systems*, Palgrave, 2003.
11. Marwedel, Peter, *Embedded System Design*, Kluwer Publishers, 2004.

CO-PO Articulation Matrix Embedded System Design Course (PEC-CSE301-T)

List of Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1. **State** the concepts related to embedded system design. (LOTS: Level 1: Remember)	1	–	–	–	–	–	–	–	–	–	–	–	3	–	–
CO2. **Discuss** the principles of embedded systems and their applications. (LOTS: Level 2: Understand)	1	–	–	–	–	–	–	–	–	–	–	–	3	–	–
CO3. **Apply** the principles of embedded design for problem solving. (LOTS: Level 3: Apply)	2	2	–	–	2	–	–	–	–	–	–	–	3	–	–
CO4. **Analyze** architectural design patterns and engineering tradeoffs. (HOTS: Level 4: Analyse)	3	3	2	2	2	–	–	–	–	–	–	–	3	–	–
CO5. **Design** architectural patterns for connected and distributed devices in the IoT. (HOTS: Level 6: Create)	3	3	2	3	3	–	–	–	–	–	–	–	3	–	–
Level of Attainments PEC-CSE3Q1-T															

Wireless and Mobile Communication

General Course Information

Course Code: PEC-CSE302-T/ PCC-IT401-T Course Credits: 3 Type: Professional Elective Contact Hours: 3 Mode: Lectures (L) Examination Duration: 3	**Course Assessment Methods (internal: 30; external: 70)** Two minor examinations (20 marks), Class Performance measured through percentage of lectures attended (4 marks), assignments (6 marks), and the end- semester examination (70 marks). For the end semester examination, nine questions are to be set by the examiner. A candidate is required to attempt 5 questions in all. All questions carry equal marks. Question number 1 will be compulsory and based on the entire syllabus. It will contain seven parts of 2 marks each. Question numbers 2 to 9 will be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt the remaining four questions by selecting one question from each of the four units.

Pre-requisites:

Basic knowledge of computer networks, Network Architecture and reference model, High Speed Network technologies, Ethernet, TCP/IP architecture.

About the course:

This course attunes the students with mobile and wireless communication using the Networking infrastructure of organizations/institutes. Students learn to analyse Networks' Architecture for wireless communication and the protocols for various layers in the Wireless Networks, technologies used and application arena of Wireless Networks.

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

CO1. **recall** different mobile and wireless communication concepts. (LOTS: Level 1: Remember)
CO2. **explain** working of different Mobile Communication Technologies used now a days. (LOTS: Level 2: Understand)
CO3. **demonstrate** application of different mobile protocols for different Mobile and Wireless Communication Technologies. (LOTS: Level 2: Understand)
CO4. **analyze** the performance of different Mobile Communication technologies in different scenarios / situations. (HOTS: Level 4: Analyse)
CO5. **design** a mobile network for any city/state/country using combination of different Mobile Technologies. (HOTS: Level 6: Create)

Course Content

Unit I

Mobile Communication:' Wireless Transmission--- Frequencies, signals, antennas, signal propagation, multiplexing, modulation, spread spectrum, cellular system. Specialized MAC, SDMA, FDMA, TDMA- fixed TDM, classical ALOHA, slotted ALOHA, CSMA, DAMA, PRMA, reservation TDMA. Collision avoidance, polling inhibit sense multiple access. CDMA, GSM- mobile services, architecture, radio interface, protocol, localization, calling, handover, security, new data services, Introduction to WLL.

Unit II

Wireless LAN IEEE 802.11-System and protocol architecture, physical layer. Frame format.
Bluetooth--- Protocol architecture, Frame format.
WiMAX – Layered Protocol architecture, frame types, format, Applications
Introduction to LTE, LTE advanced, VoLTE

Unit III

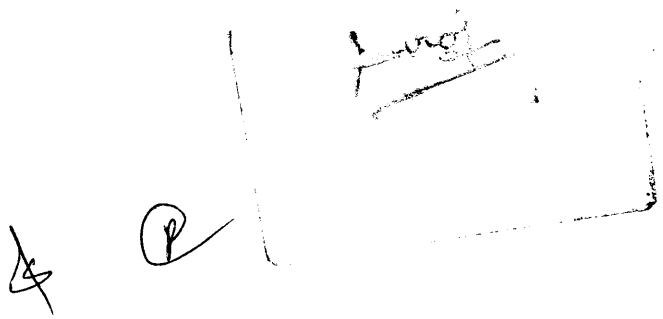
Mobile network Layer: Mobile IP- goals, assumption, requirement, entities, terminology, IP packet delivery, Agent advertisement and discovery, registration, tunneling, encapsulation, optimization , reverse tunneling, IPV6. DHCP. Adhoc Networks—routing , Destination Sequence Distance Vector, dynamic source routing, hierarchical algorithm, alternative metric.

Unit IV

Mobile Transport Layer: Traditional TCP, Indirect TCP, Snooping TCP, Mobile TCP fast retransmission/ recovery, transmission/time out freezing, selective retransmission, Transaction oriented TCP.

Text and Reference Books:

1. Jochen Schiller, *Mobile Communication*, 2nd Edition, Pearson, 2009.
2. Andrew S Tanenbaum, *Computer Networks*, 5th Edition, Pearson 2013.
3. William C Y Lee, *Mobile Communication Engineering*: *Theory and Applications*, 2nd Edition, McGraw Hill, 1997.



CO-PO Articulation Matrix Wireless and Mobile Communication Course (PEC-CSE302-T)

List of Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1. **Recall** different mobile and wireless communication concepts. (LOTS: Level 1: Remember)	1	–	–	–	–	–	–	–	–	–	–	–	–	3	–
CO2. **Explain** working of different Mobile Communication Technologies used now a days. (LOTS: Level 2: Understand)	1	–	–	–	–	–	–	–	–	–	–	–	–	3	–
CO3. **Demonstrate** application of different mobile protocols for different Mobile and Wireless Communication Technologies. (LOTS: Level 2: Understand)	1	1	–	–	–	–	–	–	–	–	–	–	–	3	–
CO4. **Analyze** the performance of different Mobile Communication technologies in different scenarios / situations. (HOTS: Level 4: Analyse)	2	2	2	2	2	–	–	–	–	–	–	–	–	3	–
CO5. **Design** a mobile network for any city/state/country using combination of different Mobile Technologies. (HOTS: Level 6: Create)	3	3	3	3	3	–	–	–	–	–	2	2	–	3	–
Level of Attainments PEC-CSE302-T															

Graph Theory

General Course Information

Course Code: PEC-CSE303-T/ PEC-IT303-T Course Credits: 3 Type: Professional Elective Contact Hours: 3 hours/week Mode: Lectures (L) Examination Duration: 3 hours	**Course Assessment Methods (internal: 30; external: 70)** Two minor examinations (20 marks), Class Performance measured through percentage of lectures attended (4 marks), assignments (6 marks), and the end- semester examination (70 marks). For the end semester examination, nine questions are to be set by the examiner. A candidate is required to attempt 5 questions in all. All questions carry equal marks. Question number 1 will be compulsory and based on the entire syllabus. It will contain seven parts of 2 marks each. Question numbers 2 to 9 will be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt the remaining four questions by selecting one question from each of the four units.

Pre-requisites: Basic knowledge of Abstract Algebra, Set Theory and Counting Techniques

About the Course:

Graph Theory is an elective course for every graduate in Computer Science and Engineering. The importance of Graph Theory reveals from the fact that it can be applied to solve any practical problem in electrical networks, operation research, data structure or social sciences etc. Also, Graph Theory provides easy representation of mathematical facts with insightful theories behind them. This course explains different types of graphical structures, related properties, various operations and facts related to these graphical structures with the help of proofs.

Course Outcomes: By the end of the course students will be able to:

CO1. **recognize** different kinds of Graphs. (LOTS: Level 1:Remember)
CO2. **demonstrate** various types of graphical structures with the operations implemented on these structures. (LOTS: Level 2: Understand)
CO3. **apply** graph theory constructs for solving problems. (LOTS: Level 3: Apply)
CO4. **justify** various facts and results associated with graphical structures with the help of proofs. (HOTS: Level 5: Evaluate)
CO5. **sketch** the graph to solve any problem in pictorial and easy representation. (HOTS: Level 6: Create)

Course Content

Unit I

Introduction to graphs, Types of graphs -Regular, Complete, Bipartite, Isomorphic, Connected, Applications, Operations on Graphs, Walks, Path, Circuits, Euler Graphs, Hamiltonian Path and Circuits, Trees, Properties of Trees, Spanning Trees (Standard Results with proofs based on all mentioned topic).

Unit II

Cut-Sets, Properties of Cut-Set, All Cut-Sets in a graph, Fundamental Circuits and Cut-Sets, Connectivity and Separability, Network Flows, 1-Isomorphism, 2- Isomorphism, Planar Graphs, Kuratowski's Two Graphs (Standard Results with proofs).

Unit III

Sets with one operation, Sets with two operations, Modular Arithmetic and Galois Fields, Vector and Vector Spaces, Vector Space associated with a graph, Basic Vectors of a graph, Circuits and Cut-Set Subspaces, Orthogonal Vectors and Spaces, Intersection and Join of W and W_s.

Unit IV

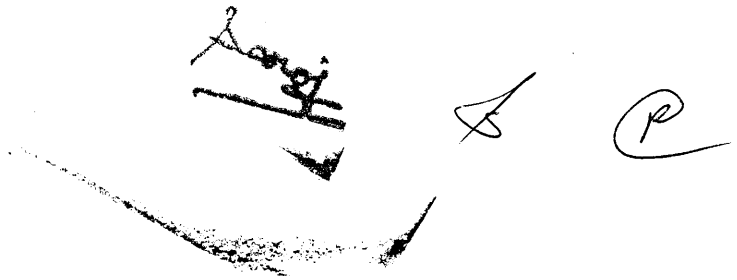
Matrix representation of graphs, Incidence Matrix, Submatrices, Circuit Matrix, Fundamental Circuit Matrix and Rank, Coloring of graphs: Chromatic Number, Vertex Coloring of graphs, Edge Coloring of graphs, Coloring of Planar Graphs.

Text and Reference Books:

1. V. K. Balakrishnan, *Graph Theory*, Tata McGraw Hill, 1st Edition, 2004.
2. Narsingh Deo, *Graph Theory with Applications to Engineering and Computer Science*, Prentice-Hall of India, Reprint, 2004.
3. Frank Harary, *Graph Theory*, Narosa/Addison Wesley, Indian Student Edition, 1988.
4. Bollobas, Bela, *Modern Graph Theory*, Springer Verlag New York, 1st Edition, 1998.
5. R. Diestel, *Graph Theory*, Springer, 2nd Edition, 2000.
6. Douglas B. West, *Introduction to Graph Theory*, Prentice Hall of India, 2nd Edition, 2002.

CO-PO Articulation Matrix Graph Theory Course (PEC-CSE303-T)

List of Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1. **Recognize** different kinds of Graphs. (LOTS: Level 1: Remember)	2	1	–	–	–	–	–	–	–	–	–	–	2	–	–
CO2. **Demonstrate** various types of graphical structures with the operations implemented on these structures. (LOTS: Level 2: Understand)	2	1	–	–	–	–	–	–	–	–	–	–	3	–	–
CO3. **Apply** graph theory constructs for solving problems. (LOTS: Level 3: Apply)	2	3	1		2	–	–	–	–	–	–	2	3	–	–
CO4. **Justify** various facts and results associated with graphical structures with the help of proofs. (HOTS: Level 5: Evaluate)	3	2	2	3	2	–	–	–	–	–	–	2	3	–	–
CO5. **Sketch** the graph to solve any problem in pictorial and easy representation. (HOTS: Level 6: Create)	3	–	1	2	2	–	–	–	–	–	–	2	3	–	–
Level of Attainments PEC-CSE303-T															



Bio-informatics

General Course Information:

Course Code: PEC-CSE304-T/ PEC-IT304-T Course Credits: 3 Type: Professional Elective Contact Hours: 3 hours/week Mode: Lectures (L) Examination Duration: 3 hours	**Course Assessment Methods (internal: 30; external: 70)** Two minor examinations (20 marks), Class Performance measured through percentage of lectures attended (4 marks), assignments (6 marks), and the end- semester examination (70 marks). For the end semester examination, nine questions are to be set by the examiner. A candidate is required to attempt 5 questions in all. All questions carry equal marks. Question number 1 will be compulsory and based on the entire syllabus. It will contain seven parts of 2 marks each. Question numbers 2 to 9 will be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt the remaining four questions by selecting one question from each of the four units.

Pre-requisites: None

About the Course:

The scope of Bio-informatics is growing rapidly. Analysing data related to bio-informatics is not possible without computational skills. This course is designed to impart fundamental knowledge of bio-informatic which would enable students to understand the intricacies of Bioinformatics. The students will learn about the characteristic of bio-informatic data and the tools for analysis of such data.

Course Outcomes: By the end of the course students will be able to:

CO1. **list** the applications of bioinformatics and biological databases. (LOTS: Level 1: Remember)
CO2. **explain** storage and retrieval of biological data from various biological databases. (LOTS: Level 2: Understand)
CO3. **apply** the knowledge of bio-informatic concepts. (LOTS: Level 3: Apply)
CO4. **identify** challenges in bioinformatics and computational biology. (HOTS: Level 4: Analyse)
CO5. **compare and contrast** various algorithms for sequence alignment and scoring algorithms. (HOTS: Level 5: Evaluate)
CO6. **devise** schemes for addressing bio-informatic problems. (LOTS: Level 6: Create)

Course Content

Unit: I

Bioinformatics: Introduction to Bioinformatics, Scope, Overview of molecular biology & genetics, Nucleic acid; structure & function, Protein structure & function; DNA Replication, Transcription, Translations, Genetic code, Codon Bias, Molecular Biology Techniques used in Bioinformatics.

Computer applications in molecular biology, Protein domains and human genome analysis program (BLAST, FASTA etc.). Search and retrieval of biological information and databases sequence, databank (NCBI)12hrs

Unit: II

Sequence Alignment

Pairwise Sequence Alignment: Evolutionary Basis, Sequence Homology versus Sequence Similarity, Sequence Similarity versus Sequence Identity, Methods, Scoring Matrices, Statistical Significance of Sequence Alignment

Database Similarity Searching: Unique Requirements of Database Searching, Heuristic Database Searching, Basic Local Alignment Search Tool (BLAST), FASTA, Comparison of FASTA and BLAST, Database Searching with the Smith–Waterman Method.

Unit: III

Multiple Sequence Alignment: Scoring Function, Exhaustive Algorithms, Heuristic Algorithms, Practical Issues.

Profiles and Hidden Markov Models: Position-Specific Scoring Matrices, Profiles, Markov Model and Hidden Markov Model.

Protein Motifs and Domain Prediction: Identification of Motifs and Domains in Multiple Sequence Alignment, Motif and Domain Databases Using Regular Expressions, Motif and Domain Databases Using Statistical Models, Protein Family Databases, Motif Discovery in Unaligned Sequences, Sequence Logos.

Unit: IV

Molecular Phylogenetics

Phylogenetics Basics: Molecular Evolution and Molecular Phylogenetics, Terminology, Gene Phylogeny versus Species Phylogeny, Forms of Tree Representation, Procedure.

Phylogenetic Tree Construction Methods and Programs: Distance-Based Methods, Character-Based Methods, Phylogenetic Tree Evaluation, Phylogenetic Programs

Text and References Books:

1. T K Attwood and D J Parry Smith , *Introduction to Bioinformatics*, Pearson Education Asia, Singapore, 2001.
2. Sensen, C.W., *Essentials of Genomics and Bioinformatics*, John Wiley and Sons. 2002
3. Attwood, T. and Pary-Smith, D., *Introduction to Bioinformatics*, Prentice Hall.1999
4. Baxevanis, A.D. and Ouellette, B.F.F., *Bioinformatics: A Practical Guide to the Analysis of genes and Protein* , Wiley- Interscience, 2001
5. Stuart M. Brown, *Bioinformatics: A Biologists Guide to Computing and the Internet*, NKU Medical Centre, NY USA, 2000.

CO-PO Articulation Matrix Bio-informatics Course (PEC-CSE304-T)

List of Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1. **List** the applications of bioinformatics and biological databases. (LOTS: Level 1: Remember)	1	–	–	–	–	–	–	–	–	–	–	–	–	–	2
CO2. **Explain** storage and retrieval of biological data from various biological databases. (LOTS: Level 2: Understand)	1	–	–	–	–	–	–	–	–	–	–	–	–	–	3
CO3. **Apply** the knowledge of bio-informatic concepts. (LOTS: Level 3: Apply)	2	–	–	–	–	–	–	–	–	–	–	–	–	–	3
CO4. **Identify** challenges in bioinformatics and computational biology. (HOTS: Level 4: Analyse)	–	2	–	–	–	–	–	–	–	–	–	–	–	–	3
CO5. **Compare and contrast** various algorithms for sequence alignment and scoring algorithms. (HOTS: Level 5: Evaluate)	2	3	2	–	2	–	–	–	–	–	–	–	–	–	3
CO6. **Devise** schemes for addressing bio-informatic problems. (LOTS: Level 6: Create)	3	3	2	3	2	–	–	–	–	–	–	–	–	–	3
Level of Attainments PEC-CSE304-T															

Operating Systems Lab. (UNIX/LINUX)

General Course Information

Course Code: PCC-CSE305-P/ PCC-IT206-P Course Credits: 1 Type: Professional Core Lab. Course Contact Hours: 2 hours/week Mode: Lab practice and assignments	**Course Assessment Methods (internal: 30; external: 70)** The internal and external assessment is based on the level of participation in lab. sessions and the timely submission of lab experiments/assignments, the quality of solutions designed for the assignments, the performance in VIVA-VOCE, the quality of lab. file and ethical practices followed. The internal examination is conducted by the course coordinator. The external examination is conducted by external examiner appointed by the Controller of Examination in association with the internal examiner appointed by the Chairperson of the Department.

Pre-requisites: Basic programming skills.

About the Course:

This lab. course on data science involves a rigorous training on R programming. It incorporates solving problems related to data science in statistical and predictive modelling framework. The objective of the lab course is to equip the students to solve the practical data science problems related to intelligent data analysis using R.

Course Outcomes: By the end of the course students will be able to:

CO1. **apply** commands related to vi and Emacs editors, general utilities and file systems. (LOTS: Level 3: Apply)
CO2. **write** basic shell scripts and use *sed* commands as well as *awk* programming. (LOTS: Level 3: Apply)
CO3. **analyse** the results of memory management and disk management commands. (HOTS: Level 4: Analyse)
CO4. **evaluate** solutions for different operating system problems such as scheduling, memory management and file management. (HOTS: Level 5: Evaluate)
CO5. **create** lab record for assignments that includes problem definitions, design of solutions and conclusions. (HOTS: Level 6: Create)
CO6. **demonstrate** use of ethical practices, self-learning and team spirit. (LOTS: Level 3: Apply)

List of experiments/assignments:

- Study of WINDOWS and Linux operating system (Linux kernel, shell, basic commands pipe & filter commands).
- Study vi editor.
- Administration of LINUX Operating System.
- Writing of Shell Scripts (Shell programming).
- AWK programming.
- Write a C program to simulate different scheduling algorithms
- Write a C program to simulate different file allocation strategies

Note:

The actual experiments/assignments will be designed by the course coordinator. One assignment should be designed to be done in groups of two or three students. The assignments must meet the objective of the course and the levels of the given course outcomes. The list of assignments and schedule of submission will be prepared by the course coordinator at the beginning of the semester.

CO-PO Articulation Matrix Operating System Lab. (PCC-CSE305-P)

List of Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1. **Apply** commands related to vi and Emacs editors, general utilities and file systems. (LOTS: Level 3: Apply)	1	–	–	–	–	–	–	–	–	–	–	–	1	–	–
CO2. **Write** basic shell scripts and use *sed* commands as well as *awk* programming. (LOTS: Level 3: Apply)	2	–	–	–	–	–	–	–	–	–	–	–	3	–	–
CO3. **Analyse** the results of memory management and disk management commands. (HOTS: Level 4: Analyse)	2	2		2	2	–	–	–	–	–	–	–	3	–	–
CO4. **Evaluate** solutions for different operating system problems such as scheduling, memory management and file management. (HOTS: Level 5: Evaluate)	2	2		2	2	–	–	–	–	–	–	–	3	–	–
CO5. **Create** lab record for assignments that includes problem definitions, design of solutions and conclusions. (HOTS: Level 6: Create)	–	–	–	–	–	–	–	–	–	3	–	–	–	–	–
CO6. **Demonstrate** use of ethical practices, self-learning and team spirit. (LOTS: Level 3: Apply)	–	–	–	–	–	–	–	3	3	–	–	3	–	–	–
Level of Attainments PCC-CSE305-P															

Data Analytics using R Lab.

General Course Information

Course Code: PCC-CSE307-P/ PEC-IT407-P Course Credits: 1.5 Type: Professional Core Lab. Course Contact Hours: 3 hours/week Mode: Lab. practice and assignments	**Course Assessment Methods (internal: 30; external: 70)** The internal and external assessment is based on the level of participation in lab. sessions and the timely submission of lab experiments/assignments, the quality of solutions designed for the assignments, the performance in VIVA-VOCE, the quality of lab. file and ethical practices followed. The internal examination is conducted by the course coordinator. The external examination is conducted by external examiner appointed by the Controller of Examination in association with the internal examiner appointed by the Chairperson of the Department.

Pre-requisites: Basic programming skills and knowledge of statistics

About the Course:

This lab. course on data science involves a rigorous training on R programming. It incorporates solving problems related to data science in statistical and predictive modelling framework. The objective of the lab course is to equip the students to solve the practical data science problems related to intelligent data analysis using R.

Course Outcomes: By the end of the course students will be able to:

CO1. **implement** R programming concepts for data analysis. (LOTS: Level 3: Apply)
CO2. **analyse** the trends in data through exploratory data analysis. (HOTS: Level 4: Analyse)
CO3. **evaluate** the results of descriptive and inferential statistics. (HOTS: Level 5: Evaluate)
CO4. **devise** solutions for descriptive and predictive modelling. (HOTS: Level 6: Create)
CO5. **create** lab. Record of assignment solutions that include problem definition, solutions and interpretation of results. (HOTS: Level 6: Create)
CO6. **demonstrate** use of ethical practices, independent enquiry and self-learning, and team spirit to solve unseen problems. (LOTS: Level 3: Apply)

List of experiments/assignments

1. Install R studio and explore its GUI. Explore the base R package- datasets. See the list of datasets available in the package. Write description for the following datasets:
   i. HairEyeColor
   ii. Iris
   iii. Airquality
   iv. mtcars

   In addition to general description of the dataset, it should include the number of attributes and instances, class of the datasets. It should also include the type of each attribute. Apply *summary()* and *str()* functions to these datasets.
2. Three assignment related to creating and manipulating objects like vectors, factors, matrices, lists and data frames.

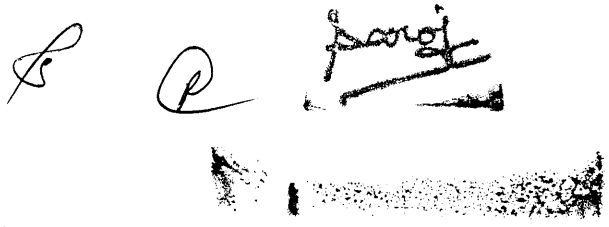
3. Two assignments on the use of control, looping statements and user defined functions.
4. Two assignment on finding descriptive statistics and exploratory data analysis.
5. Two assignments on making different charts and writing the finding on the basis of these charts.
6. Two assignments on hypothesis testing for descriptive and inferential statistics.
7. Two assignments on predictive modelling using R packages in groups of two or three students depending on the size of the assignment.

Note:

The actual experiments/assignments will be designed by the course coordinator. One assignment should be designed to be done in groups of two or three students. The assignments must meet the objective of the course and the levels of the given course outcomes. The list of assignments and schedule of submission will be prepared by the course coordinator at the beginning of the semester.

CO-PO Articulation Matrix Data Analytics using R Lab. (PCC- CSE307-P)

List of Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1. **Implement** R programming concepts for data analysis. (LOTS: Level 3: Apply)	2	1	-	-	3	–	–	–	–	–	–	–	–	–	2
CO2. **Analyse** the trends in data through exploratory data analysis. (HOTS: Level 4: Analyse)	2	2	-	3	3	–	–	–	–	–	–	–	–	–	3
CO3. **Evaluate** the results of descriptive and inferential statistics. (HOTS: Level 5: Evaluate)	3	3	-	3	3	–	–	–	–	–	–	–	–	–	3
CO4. **Devise** solutions for descriptive and predictive modelling. (HOTS: Level 6: Create)	3	3	1	3	3	–	–	–	–	–	–	–	–	–	3
CO5. **Create** lab. Record of assignment solutions that include problem definition, solutions and interpretation of results. (HOTS: Level 6: Create)	–	–	–	–	–	–	–	–	–	3	–	–	–	–	–
CO6. **Demonstrate** use of ethical practices, independent enquiry and self-learning, and team spirit to solve unseen problems. (LOTS: Level 3: Apply)	–	–	–	–	–	–	–	3	3	–	–	3	–	–	–
Level of Attainments PCC-CSE307-P															



.NET using C# Lab.

General Course Information

Course Code: PCC-CSE308-P PCC-IT302-P Course Credits: 1 Type: Professional Core Lab. Course Contact Hours: 2 hours/week Mode: Lab practice and assignments	**Course Assessment Methods (internal: 30; external: 70)** The internal and external assessment is based on the level of participation in lab. sessions and the timely submission of lab experiments/assignments, the quality of solutions designed for the assignments, the performance in VIVA-VOCE, the quality of lab. file and ethical practices followed. The internal examination is conducted by the course coordinator. The external examination is conducted by external examiner appointed by the Controller of Examination in association with the internal examiner appointed by the Chairperson of the Department.

Pre-requisites: Programming knowledge of C++ and HTML basics

About the Course:

This lab course involves implementation of basic and advanced programs of C#. The objective of the lab. course is to train the students to solve the problems related to Object Oriented Technology, ADO.NET Connectivity and Web Applications using XML.

Course Outcomes: By the end of the lab course a student would be able to:

CO1. **implement** C# programs in .NET framework. (LOTS: Level 3: Apply)
CO2. **apply** ADO.NET for developing database applications. (LOTS: Level 3: Apply)
CO3. **analyse** given programs for their correctness and efficiency for given inputs and expected outputs. (HOTS: Level 4: Analysis)
CO4. **integrate** HTML code with ASP.NET and HTML code for designing a web pages. (HOTS: Level 6: Create)
CO5. **create** written records for the given assignments with problem definition, design of solution and conclusions. (HOTS: Level 6: Create)
CO6. **demonstrate** ethical practices while solving problems individually or in groups (LOTS: Level 3: Apply).

List of experiments/assignments

1. Write a console application that obtains four int values from the user and displays the product.
2. Write an application that receives the following information from a set of students:

   Student Id:

   Student Name:

   Course Name:

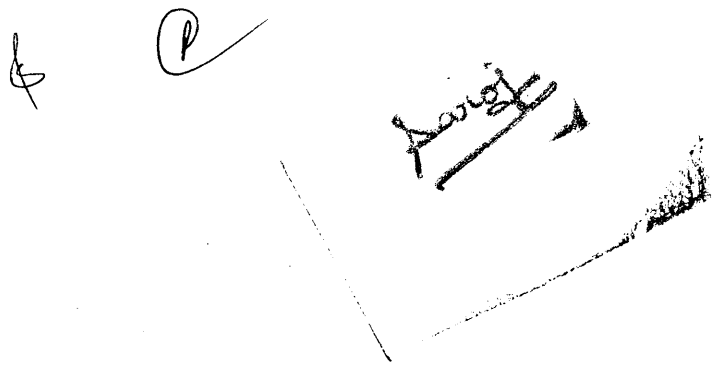
   Date of Birth:

   The application should also display the information of all the students once the data is Entered. Implement this using an Array of Structures.

3. Database programs with ASP.NET and ADO.NET Create a Login Module which adds Username and Password in the database. Username in the database should be a primary key.
4. Create a web application to insert 3 records inside the SQL database table having following fields (DeptId, DeptName, EmpName, Salary). Update the salary for any one employee and increment it to 15% of the present salary. Perform delete operation on 1 row of the database table.
5. Create a web page to display the cricket score from the table event(id, name, score). Refresh the website automatically after every 30 seconds.
6. Write a C# Sharp program to extract the Date property and display the Date Time value in the formatted output
7. Write a program in C# Sharp to count a total number of alphabets, digits and special characters in a string.
8. Create a web page to display animation using JQuery.
9. Create a web page to display hide, show, slidedown, slideup and Toggle effects for paragraph tags, using JQuery

Note:

The actual experiments/assignments will be designed by the course coordinator. One assignment should be designed to be done in groups of two or three students. The assignments must meet the objective of the course and the levels of the given course outcomes. The list of assignments and schedule of submission will be prepared by the course coordinator at the beginning of the semester.



CO-PO Articulation Matrix .NET using C# Lab. (PCC- CSE308-P)

List of Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1. **Implement** C# programs in .NET framework. (LOTS: Level 3: Apply)	2	1	-	-	3	–	–	–	–	–	–	–	–	–	2
CO2. **Apply** ADO.NET for developing database applications. (LOTS: Level 3: Apply)	2	2	-	3	3	–	–	–	–	–	–	–	–	–	3
CO3. **Analyse** given programs for their correctness and efficiency for given inputs and expected outputs. (HOTS: Level 4: Analysis)	3	3	-	3	3	–	–	–	–	–	–	–	–	–	3
CO4. **Integrate** HTML code with ASP.NET and HTML code for designing web pages. (HOTS: Level 6: Create)	3	3	1	3	3	–	–	–	–	–	–	–	–	–	3
CO5. **Create** written records for the given assignments with problem definition, design of solution and conclusions. (HOTS: Level 6: Create)	–	–	–	–	–	–	–	–	–	3	–	–	–	–	–
CO6. **Demonstrate** ethical practices while solving problems individually or in groups (LOTS: Level 3: Apply).	–	–	–	–	–	–	–	3	3	–	–	3	–	–	–
Level of Attainments PCC-CSE308-P															

Compiler Design

General Course Information

Course Code: PCC-CSE401-T/ PCC-IT306-T Course Credits: 3 Type: Professional Core Contact Hours: 3 hours/week Mode: Lectures (L) Examination Duration: 3 hours	**Course Assessment Methods (internal: 30; external: 70)** Two minor examinations (20 marks), Class Performance measured through percentage of lectures attended (4 marks), assignments (6 marks), and the end- semester examination (70 marks). For the end semester examination, nine questions are to be set by the examiner. A candidate is required to attempt 5 questions in all. All questions carry equal marks. Question number 1 will be compulsory and based on the entire syllabus. It will contain seven parts of 2 marks each. Question numbers 2 to 9 will be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt the remaining four questions by selecting one question from each of the four units.

Pre-requisites: Brief knowledge of programming languages, Data Structure, and Algorithm Design.

About the Course:

Compilers have become part and parcel of today's computer systems. These are responsible for making the user's computing requirements, specified as a piece of program, understandable to the underlying machine. There tools work as interface between the entities of two different domains – the human being and the machine. The actual process involved in this transformation is quite complex. Compiler design covers basic translation mechanism and, error detection and recovery. It includes lexical, syntax, and semantic analysis as front end, and code generation and optimization as back-end.

Course Outcomes: By the end of the course students will be able to:

CO1. **state** principles of compiler design. (LOTS: Level 1: Remember)
CO2. **illustrate** the essential phases for automatically converting source code into object code. (LOTS: Level 2: Understand)
CO3. **apply** lexical analysis, syntax analysis and code optimization techniques for solving problems. (LOTS: Level 3: Apply)
CO4. **analyse** a parse tree and a given BNF grammar. (LOTS: Level 4: Analyse)
CO5. **compare and contrast** syntax-oriented translation schemes (HOTS: Level 5: Evaluate)
CO6. **design** a lexical analyser from the specification of a language's lexical rules. (HOTS: Level 6: Create)

Course Content

Unit I

Introduction To Compilers: Compilers and translators, need of translators, structure of compiler its different phases, Compiler construction tools.

Lexical Analysis: Role of lexical analyzer, design of lexical analyzer, regular expressions, Specification and recognition of tokens, input buffering, A language specifying lexical analyzer. Finite automata, conversion from regular expression to finite automata, and vice versa, minimizing number of states of DFA, Implementation of lexical analyzer.

Unit II

Syntax Analysis: Role of parsers, context free grammars, definition of parsing. Parsing Technique: Shift-reduce parsing, operator precedence parsing, top down parsing, predictive parsing.

Unit III

LR parsers, SLR, LALR and Canonical LR parser. Syntax Directed Translations: Syntax directed definition, construction of syntax trees, syntax directed translation scheme, implementation of syntax directed translation, three address code, quadruples and triples.

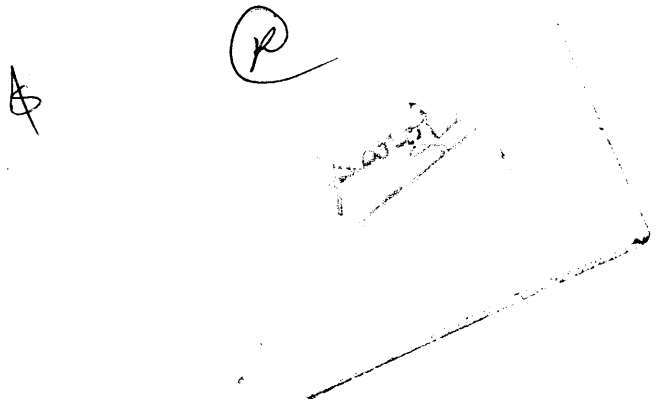
Unit IV

Symbol Table & Error Detection and Recovery: Symbol tables, its contents and data structure for symbol tables; trees, arrays, linked lists, hash tables. Errors, lexical phase error, syntactic phase error, semantic error.

Code Optimization & Code Generation: Code generation, forms of objects code, machine dependent code, optimization, register allocation for temporary and user defined variables.

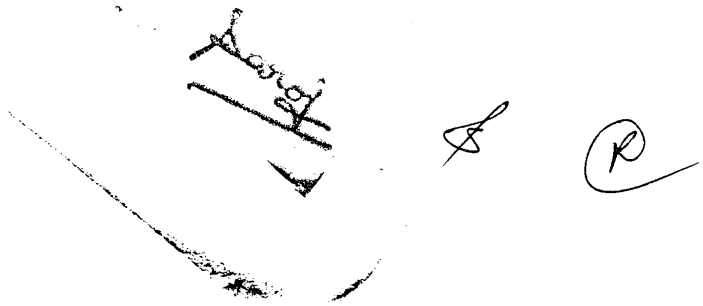
Text and Reference Books:

1. Alfread V. AHO, Ravi Sethi and J.D. Ullman, *Compilers Principle, Techniques and Tools*, Addison Wesley, 2007.
2. Tremblay and Sorenson, *Theory and practice of compiler writing*, Mc. Graw Hill, 1985.
3. Dhamdare, *System software*, MGH, 1986.
4. Alfred V. Aho, Jeffrey D. Ullman, *Principles of Compiler Design*, Narosa Publication, 2002.



CO-PO Articulation Matrix Compiler Design Course (PCC-CSE401-T)

List of Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1. **State** principles of compiler design. (LOTS: Level 1: Remember)	1	–	–	–	–	–	–	–	–	–	–	–	2	–	–
CO2. **Illustrate** the essential phases for automatically converting source code into object code. (LOTS: Level 2: Understand)	1	–	–	–	–	–	–	–	–	–	–	–	3	–	–
CO3. **Apply** lexical analysis, syntax analysis and code optimization techniques for solving problems. (LOTS: Level 3: Apply)	2	1	–	–	–	–	–	–	–	–	–	–	3	–	–
CO4. **Analyse** a parse tree and a given BNF grammar. (LOTS: Level 4: Analyse)	3	2	1		2	–	–	–	–	–	–	–	3	–	–
CO5. **Compare and contrast** syntax-oriented translation schemes (HOTS: Level 5: Evaluate)	2	2	1		2	–	–	–	–	–	–	–	3	–	–
CO6. **Design** a lexical analyser from the specification of a language's lexical rules. (HOTS: Level 6: Create)	3	3	2	2	3	–	–	–	–	–	–	–	3	–	–
Level of Attainments PCC-CSE-401-T														–	–



Artificial Intelligence

General Course Information

Course Code: PCC-CSE402-T/ PCC-IT304-T Course Credits: 3 Type: Professional Core Contact Hours: 3hours/week Mode: Lectures (L) Examination Duration: 3 hours	**Course Assessment Methods (internal: 30; external: 70)** Two minor examinations (20 marks), Class Performance measured through percentage of lectures attended (4 marks), assignments (6 marks), and the end- semester examination (70 marks). For the end semester examination, nine questions are to be set by the examiner. A candidate is required to attempt 5 questions in all. All questions carry equal marks. Question number 1 will be compulsory and based on the entire syllabus. It will contain seven parts of 2 marks each. Question numbers 2 to 9 will be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt the remaining four questions by selecting one question from each of the four units.

Pre-requisites: Basic knowledge of Algorithms and probability.

About the Course:

Artificial Intelligence is a core and an essential course for every graduate in Computer Science and Engineering. This course introduces the concepts of Artificial Intelligence and challenges inherent in building intelligent systems. It includes the role of knowledge representation in problem solving and how these are used in making intelligent machine. Further it incorporates the concepts of expert system and its applications.

Course Outcomes: By the end of the course students will be able to:

CO1. **outline** various Artificial Intelligence techniques. (LOTS: Level 1: Remember)
CO2. **illustrate** reasoning under uncertainty. (LOTS: Level 2: Understand)
CO3. **apply** search and knowledge representation techniques to solve AI problems.(LOTS: Level 3: Apply)
CO4. **compare** strengths and weaknesses of AI algorithms (HOTS: Level 4: Analyse).
CO5. **combine** various AI techniques to solve intelligent systems' problems. (HOTS: Level 6: Create)

Course Content

Unit I

Introduction to AI: Introduction, Turing Test, AI problems, State Space Search, production system

Problem Solving Using Search: Blind search techniques - Breadth first search, Depth first search. Heuristic search techniques - Generate and test, Hill Climbing, Best first search, A* Algorithm, AO* Algorithm, The Minimax Search Procedure, Adding Alpha-Beta Cut-offs.

Unit II

Knowledge Representation: Introduction, Knowledge Representation- Representation and Mappings, Symbolic Logic - Propositional logic, Predicate logic- Representing simple facts in logic, Representing Instances and ISA Relationship, Computable functions and Predicates, Unification, Resolution.

Representing Knowledge Using Rules: Procedural versus Declarative Knowledge, Logic Programming, Forward versus Backward Reasoning, Matching, Control Knowledge.

Unit III

Reasoning Under Uncertainty: Introduction to Nonmonotonic Reasoning, Probability and Baye's Theorem, Certainty Factors and Rule-based Systems, Bayesian Networks.

Fuzzy logic system: Introduction, Crisp Set, Fuzzy Sets, Fuzzy Membership Functions, Operations on Fuzzy Sets, Fuzzy Relations.

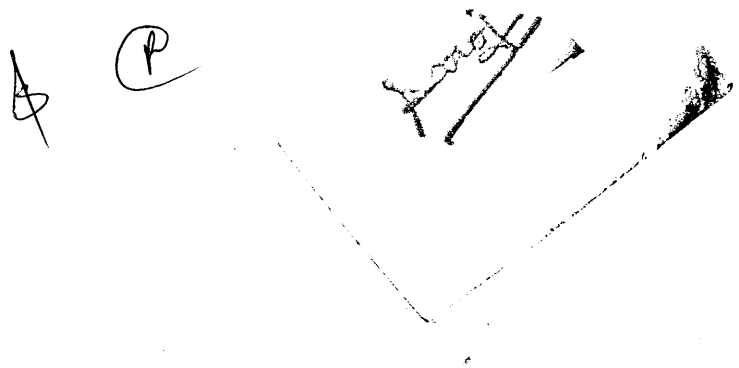
Unit IV

Planning: Introduction, Components of Planning System, Goal Stack Planning, Nonlinear Planning using Constraint Posting, Hierarchical Planning.

Expert System and Applications: Introduction, Architecture, Rule based Expert Systems, Applications of Expert Systems.

Text and Reference Books:

1. Elaine Rich, Kevin Knight and Shivashankar B Nair, *Artificial intelligence*, McGraw Hill Education. 3rd edition, 2009.
2. Stuart Russel and Peter Norvig, *Artificial intelligence: A modern Approach*, Pearson Education, 3rd edition, 2015.
3. Dan W. Patterson, *Introduction to Artificial Intelligence and Expert System*, Pearson Education.1st edition, 2007.
4. Deepak Khemani, *A first course in Artificial Intelligence*, McGraw Hill Education. 3rd edition, 1st edition, 2013.
5. George F. Luger, *Artificial Intelligence: Structures and Strategies for Complex Problem Solving*, Pearson Education, 5th edition, 2009.



CO-PO Articulation Matrix Artificial Intelligence Course (PCC-CSE402-T)

List of Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1. **Outline** various Artificial Intelligence techniques. (LOTS: Level 1: Remember)	1	–	–	–	–	–	–	–	–	–	–	–	–	–	3
CO2. **Illustrate** reasoning under uncertainty. (LOTS: Level 2: Understand)	1	–	–	–	–	–	–	–	–	–	–	–	–	–	3
CO3. **Apply** search and knowledge representation techniques to solve AI problems. (LOTS: Level 3: Apply)	2	2	–	2	2	–	–	–	–	–	–	–	–	–	3
CO4. **Compare** strengths and weaknesses of AI algorithms (HOTS: Level 4: Analyse).	2	2	2	2	–	–	–	–	–	–	–	–	–	–	3
CO5. **Combine** various AI techniques to solve intelligent systems' problems. (HOTS: Level 6: Create)	3	3	3	3	2	2	–	–	–	–	–	3	–	–	3
Level of Attainments PCC-CSE402-T															

Software Project Management

General Course Information

Course Code: PEC-CSE401-T/ PEC-IT401-T Course Credits: 3 Type: Professional Elective Contact Hours: 3 hours Mode: Lectures (L) Examination Duration: 3 hours	**Course Assessment Methods (internal: 30; external: 70)** Two minor examinations (20 marks), Class Performance measured through percentage of lectures attended (4 marks), assignments (6 marks), and the end- semester examination (70 marks). For the end semester examination, nine questions are to be set by the examiner. A candidate is required to attempt 5 questions in all. All questions carry equal marks. Question number 1 will be compulsory and based on the entire syllabus. It will contain seven parts of 2 marks each. Question numbers 2 to 9 will be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt the remaining four questions by selecting one question from each of the four units.

Pre-requisites: Preliminary knowledge of Software Engineering.

About the Course:

The course involves training students in software project management and project planning. It focuses on the need for careful planning, monitoring and control for delivering quality projects in time. Besides this student learn to measure the success of a project in meeting its objectives.

Course Outcomes: By the end of the course students will be able to:

CO1. **outline** basic concepts related to stepwise project planning. (LOTS: Level 1: Remember)
CO2. **demonstrate** the knowledge about Quality Control, Standard and Risk Management. (LOTS: Level 2: Understand)
CO3. **illustrate** the Activity Planning, and Resource Allocation Process. (LOTS: Level 2: Understand)
CO4. **apply** the concept of team structure and organization structure. (LOTS: Level 3: Apply)
CO5. **compare** various Project Evaluation and Estimation Techniques. (HOTS: Level 4: Analyse)
CO6. **plan** activities necessary for completing the software projects successfully. (HOTS: Level 6: Create)

Course Content

Unit I

Introduction to Software Project Management(SPM): Definition of Software Project, Software Project Vs Other types of projects, activities covered by SPM, categorizing software projects, project as system, management control, Requirement specification, Information and control in organization, project management lifecycle.

Stepwise Project Planning: Introduction, selecting a project, identifying project scope and objectives, identifying project infrastructure, analysing project characteristics, identifying the project products and activities, estimate efforts for each activity, identifying activity risk, allocate resources, review/publicize plan.

Unit II

Project Evaluation and Estimation: Cost-Benefit analysis, cash flow forecasting, cost benefit evaluation techniques, Selection of an appropriate project, choosing technologies, choice of process models, rapid application development, waterfall model, V process model and spiral model, Albrecht function point analysis.

Activity Planning: Objectives of activity planning, project schedule, projects and activities, sequencing and scheduling activities, network planning model.

Unit III

Risk Management: Introduction, the nature of risk, managing risk, risk identification, risk analysis, reducing the risks, evaluating risks to schedule, calculating z-values.

Resource Allocation: Introduction, the nature of resources, identifying resource requirements, scheduling resources, creating critical paths.

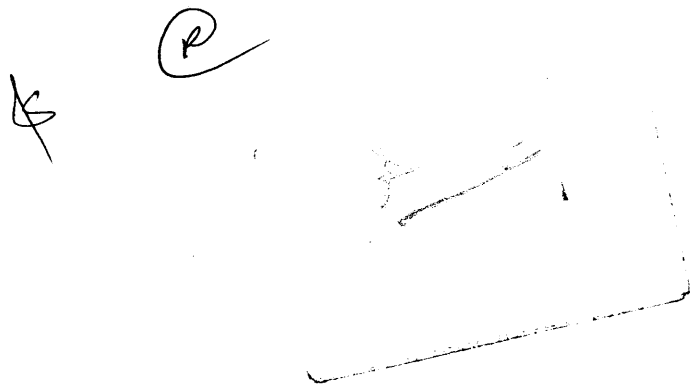
Unit IV

Managing Contracts and People: Introduction, types of contract, stages in contract placement, terms of contract, contract management, acceptance, managing people and organizing teams: Introduction, understanding organization behaviour: a back ground, selecting the right person for job, instruction in best methods, motivation, working in groups, becoming a team, decision making, leadership, organization structures.

Software Quality: Introduction, the place of software quality in project planning, the importance of software quality, defining software quality, McCall's software quality factors, product versus process quality management, external standards, techniques to enhance software quality.

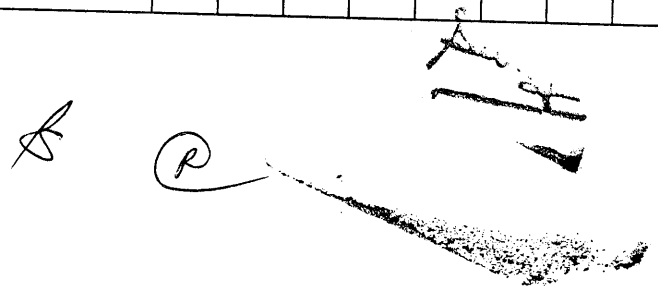
Text and Reference Books:

1. Bob Hughes and Mike Cotterell , *Software Project Management*, Sixth Edition, TMH, 2018.
2. Walker Royce , *Software Project Management*, , Addison Wesley, 1998.
3. Pankaj Jalote , *Software Project Management in Practice*, Pearson, 2002.
4. Ramesh, *Managing Global Software Projects*, TMH, 2005.



CO-PO Articulation Matrix Software Project Management Course (PEC-CSE401-T)

List of Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1. **Outline** basic concepts related to stepwise project planning. (LOTS: Level 1: Remember)	1	–	–	–	–	–	–	–	–	–	–	–	2	–	–
CO2. **Demonstrate** the knowledge about Quality Control, Standard and Risk Management. (LOTS: Level 2: Understand)	1	–	–	–	–	–	–	–	–	–	–	–	3	–	–
CO3. **Illustrate** the Activity Planning, and Resource Allocation Process. (LOTS: Level 2: Understand)	1	–	–	–	2	–	–	–	–	–	–	–	3	–	–
CO4. **Apply** the concept of team structure and organization structure. (LOTS: Level 3: Apply)	2	2	–	–	2	–	–	–	2	–	–	–	3	–	–
CO5. **Compare** various Project Evaluation and Estimation Techniques. (HOTS: Level 4: Analyse)	2	2	2	2	2	–	–	–	–		2	–	3	–	–
CO6. **Plan** activities necessary for completing the software projects successfully. (HOTS: Level 6: Create)	3	3	–	3	3	–	–	–	–	2	3	2	3	–	–
Level of Attainments PEC-CSE401-T															



Soft Computing

General Course Information

Course Code: PEC-CSE402-T/ PEC-IT302-T Course Credits: 3 Type: Professional Elective Contact Hours: 3 hours/week Mode: Lectures (L) Examination Duration: 3 hours	**Course Assessment Methods (internal: 30; external: 70)** Two minor examinations (20 marks), Class Performance measured through percentage of lectures attended (4 marks), assignments (6 marks), and the end- semester examination (70 marks). For the end semester examination, nine questions are to be set by the examiner. A candidate is required to attempt 5 questions in all. All questions carry equal marks. Question number 1 will be compulsory and based on the entire syllabus. It will contain seven parts of 2 marks each. Question numbers 2 to 9 will be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt the remaining four questions by selecting one question from each of the four units.

Pre-requisites: Basic knowledge of Probability Theory, Set Theory and, Data Structure and Computer Algorithms

About the Course:

We need to learn soft computing techniques to make intelligent machines that possess human like abilities to reason, learn and handle the uncertainty and vagueness often inherent in real world problems. Unlike conventional computing, soft computing techniques are tolerant of imprecision, uncertainty and approximations, and provide low cost, robust and tractable solutions to the complex real-world problems where conventional methods fail to do so. This introductory course on soft computing is going to cover Genetic Algorithms, Artificial Neural Networks and Fuzzy Logic.

Course Outcomes: By the end of the course students will be able to:

CO1. **define** the terminology and concepts related to soft computing techniques. (LOTS: Level 1: Remember)
CO2. **discuss** soft computing techniques including genetic algorithms, fuzzy systems and neural networks. (LOTS: Level 2: Understand)
CO3. **solve** problems related to Genetic algorithms, Fuzzy logic and Neural Networks. (LOTS: Level 3: Apply)
CO4. **analyse** the design of Genetic Algorithms, Neural Networks and Fuzzy Systems. (HOTS: Level 4: Analyse)
CO5. **justify** the design of a soft computing algorithm for a given problem. (HOTS: Level 5: Evaluate)
CO6. **design** Genetic Algorithms and Neural Networks to solve optimization and pattern recognition problems. (HOTS: Level 6: Create)

Course Content

Unit I

Introduction to Soft Computing and related definitions: Defining soft computing, Differentiating the situations for application of hard and soft computing; Working of a simple Genetic Algorithm: Representation/Encoding Schemes, initializing a GA population, evaluation function, genetic operators, Function optimization using GA. Study of parameters of genetic algorithms and its performance, sampling and selection mechanisms, Scaling of GA population.

Unit II

Designing Genetic Algorithms for different applications: Different types encoding schemes, role of fitness function, different types of genetic operators, Designing GAs for numerical optimization, knapsack problem and travelling salesperson and other similar problems.

Unit III

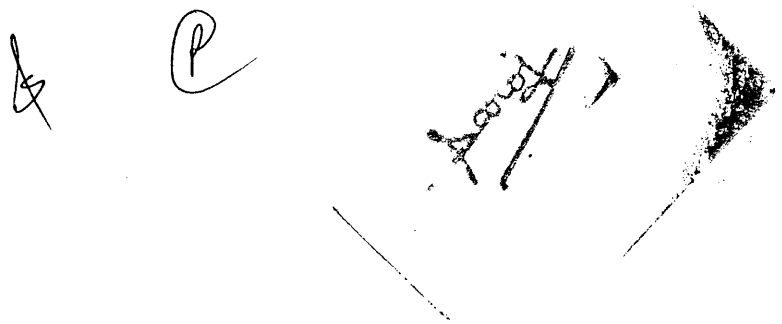
Fuzzy sets: Basic terminology and definitions, Operations on Fuzzy sets, MF formulations and parameterisation, MFs of one and two dimensions, Derivatives of parameterised MFs, Fuzzy numbers, Extension principle and fuzzy relations, Operations on Fuzzy relations, Linguistic variables, Fuzzy If-Then Rules, Compositional rule of inference.

Unit IV

Neural networks: Basic terminology and definitions, Model of an artificial neuron, Sigmoid function, Neural Network Architectures, Rosenblatt's Perceptron, Fixed increment perceptron learning algorithm for a classification problem, Examples of learning of AND/OR gate by perceptron, XOR problem. Back Propagation Neural Networks: Architecture of a backpropagation network, Model for multi-layer perceptron, Back propagation learning, Delta or gradient descent learning rule and effect of learning rate, Back propagation learning algorithm.

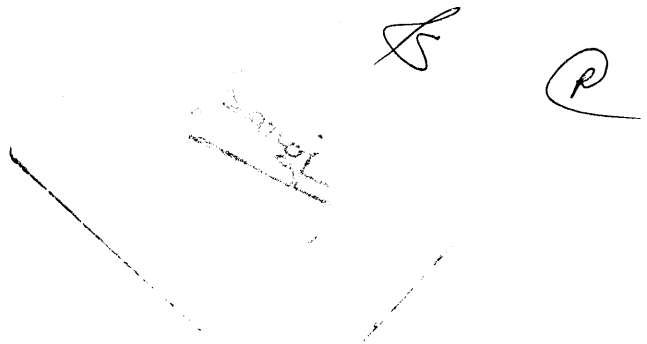
Text and Reference Books:

1. David. E. Goldberg, *Genetic Algorithms in Search, Optimization and machine learning*, Addison Wesley, 1999.
2. Zbigniew Michalewicz, *Genetic algorithms + Data Structures = Evolution Programs*, Springers-Verlag, 1999.
3. M. Mitchell, *An Introduction to Genetic Algorithms*, Prentice-Hall, 1998.
4. S. Rajasekaran & G. A. Vijayalakshmi Pai, *Neural Networks, Fuzzy Logic and Genetic Algorithms: Synthesis & Applications*, PHI, 2003.
5. S. N. Sivanandam & S. N. Deepa, *Principles of Soft Computing*, Wiley - India, 2007.
6. J-S. R. Jang, C.-T. Sun, E. Mizutani, *Neuro-Fuzzy and Soft Computing*, PHI, 1997.
7. Simon O. Haykin, *Neural Networks, A Comprehensive Foundation*, PHI, 1994.



CO-PO Articulation Matrix Soft Computing Course (PEC-CSE402-T)

List of Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1. **Define** the terminology and concepts related to soft computing techniques. (LOTS: Level 1: Remember)	1	–	–	–	–	–	–	–	–	–	–	–	–	–	3
CO2. **Discuss** soft computing techniques including genetic algorithms, fuzzy systems and neural networks. (LOTS: Level 2: Understand)	1	–	–	–	–	–	–	–	–	–	–	–	–	–	3
CO3. **Solve** problems related to Genetic algorithms, Fuzzy logic and Neural Networks. (LOTS: Level 3: Apply)	3	3	–	–	2	–	–	–	–	–	–	–	–	–	3
CO4. **Analyse** the design of Genetic Algorithms, Neural Networks and Fuzzy Systems. (HOTS: Level 4: Analyse)	3	3	–	2	2	–	–	–	–	–	–	–	–	–	3
CO5. **Justify** the design of a soft computing algorithm for a given problem. (HOTS: Level 5: Evaluate)	3	3	2	2	–	–	–	–	–	–	–	–	–	–	3
CO6. **Design** Genetic Algorithms and Neural Networks to solve optimization and pattern recognition problems. (HOTS: Level 6: Create)	3	3	3	3	3	–	–	–	–	–	–	–	–	–	3
Level of Attainments PEC-CSE402-T															



Distributed Operating System

General Course Information

Course Code: PEC-CSE-403-T/ PEC-IT403-T Course Credits: 3 Type: Professional Elective Contact Hours: 3 hours/week Mode: Lectures (L) Examination Duration: 03 hours	**Course Assessment Methods (internal: 30; external: 70)** Two minor examinations (20 marks), Class Performance measured through percentage of lectures attended (4 marks), assignments (6 marks), and the end- semester examination (70 marks). For the end semester examination, nine questions are to be set by the examiner. A candidate is required to attempt 5 questions in all. All questions carry equal marks. Question number 1 will be compulsory and based on the entire syllabus. It will contain seven parts of 2 marks each. Question numbers 2 to 9 will be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt the remaining four questions by selecting one question from each of the four units.

Pre-requisites: Knowledge of operating system, computer networks and a programming language

About the Course:

This course focuses on the study of distributed system concepts and its applications. In this course various advantages of distributed computing system are studied. After studying this course, a student will be expected to understand the design issues of the distributed operating systems and propose solutions for problems specific to the domain.

Course Outcomes: By the end of the course students will be able to:

CO1. **state** the basic concepts of distributed systems and their advantages over simple client server based computer networks. (LOTS: Level 1: Remember)

CO2. **explain** strategies for synchronization, scheduling policies and deadlock avoidance in distributed environment. (LOTS: Level 2: Understand)

CO3. **apply** distributed operating system's concepts to solve the problems inherent in distributed systems. (LOTS: Level 3: Apply)

CO4. **analyse** trends in distributed file systems. (HOTS: Level 4: Analyse)

CO5. **compare** and **contrast** strategies for synchronization, scheduling policies and deadlock avoidance and distributed file systems. (HOTS: Level 5: Evaluate)

Course Content

Unit I

Introduction: Introduction to distributed system, Goals of distributed system, Hardware and Software concepts, Design issues, Communication in distributed system: Layered protocols, ATM networks, Client- Server model, Remote Procedure Calls and Group Communication, Middleware and Distributed Operating Systems.

Unit II

Synchronization in Distributed System: Clock synchronization, Mutual Exclusion, Election algorithm, Bully algorithm, Ring algorithm, Atomic Transactions, Deadlock in Distributed Systems, Distributed Deadlock Prevention, Distributed Deadlock Detection.

Unit-III

Processes and Processors in distributed systems: Threads, System models, Processors Allocation, Scheduling in Distributed System, Real Time Distributed Systems.

Unit IV

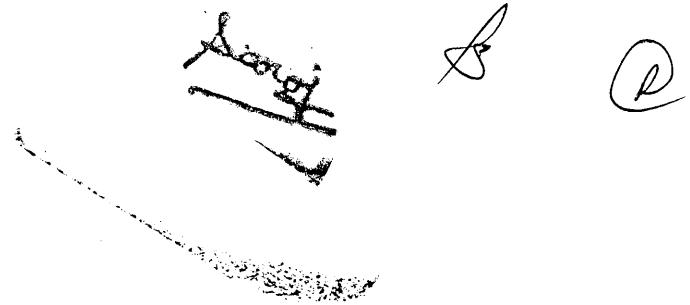
Distributed file systems: Distributed file system design, Distributed file system Implementation, Trends in Distributed file systems. Distributed Shared Memory: What is shared memory, Consistency models, Page based distributed shared memory, shared variables distributed shared memory.

Text and Reference Books:

1. Tanenbaum A.S., Van Steen M., *Distributed Systems: Principles and Paradigms*, Pearson Education,
2. Pradeep K Sinha, *Distributed Operating Systems: Concepts and Design*, Prentice Hall of India, 2007.
3. Liu M.L., *Distributed Computing, Principles and Applications*, Pearson Education, 2004.
4. Nancy A Lynch, *Distributed Algorithms*, Morgan Kaufman Publishers, USA, 2003.

CO-PO Articulation Matrix Distributed Operating System Course (PEC-CSE403-T)

List of Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1. **State** the basic concepts of distributed systems and their advantages over simple client server-based computer networks. (LOTS: Level 1: Remember)	1	–	–	–	–	–	–	–	–	–	–	–	2	–	–
CO2. **Explain** strategies for synchronization, scheduling policies and deadlock avoidance in distributed environment. (LOTS: Level 2: Understand)	1	–	–	–	–	–	–	–	–	–	–	–	3	–	–
CO3. **Apply** distributed operating system's concepts to solve the problems inherent in distributed systems. (LOTS: Level 3: Apply)	2	1	–	–	–	–	–	–	–	–	–	–	3	–	–
CO4. **Analyse** trends in distributed file systems. (HOTS: Level 4: Analyse)	2	2	2	2	–	–	–	–	–	–	–	–	3	–	–
CO5. **Compare** and **contrast** strategies for synchronization, scheduling policies and deadlock avoidance and distributed file systems. (HOTS: Level 5: Evaluate)	2	2	3	3	–	–	–	–	–	–	–	–	3	–	–
Level of Attainments PEC-CSE403-T															



Cloud Computing

General Course Information

Course Code: PEC-CSE404-T/ PEC-IT-404-T Course Credits: 3 Type: Professional Elective Contact Hours: 3hours/week Mode: Lectures (L) Examination Duration: 3 hours	**Course Assessment Methods (internal: 30; external: 70)** Two minor examinations (20 marks), Class Performance measured through percentage of lectures attended (4 marks), assignments (6 marks), and the end- semester examination (70 marks). For the end semester examination, nine questions are to be set by the examiner. A candidate is required to attempt 5 questions in all. All questions carry equal marks. Question number 1 will be compulsory and based on the entire syllabus. It will contain seven parts of 2 marks each. Question numbers 2 to 9 will be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt the remaining four questions by selecting one question from each of the four units.

Pre-requisites: Basics of Computer Network, Distributed System.

About the Course:

The objective of the course is to give students a comprehensive view of storage and networking infrastructures for highly virtualized cloud ready deployments. The course discusses the concepts and features related to Virtualized data-centre and cloud, information storage and design of applications.

Course Outcomes: By the end of the course students will be able to:

CO1. **define** concepts related to cloud computing. (LOTS: Level 1: Remember)
CO2. **express** deployment models for clouds. (LOTS: Level 2: Understand)
CO3. **apply** cloud computing techniques for various applications. (LOTS: Level 3: Apply)
CO4. **analyse** cloud computing services used at various levels. (HOTS: Level 4: Analyse)
CO5. **assess** real time cloud services. (HOTS: Level 5: Evaluate)

Course Content

Unit I

Introduction: Distributed Computing, Cluster Computing, Grid Computing, Overview of Cloud Computing, History of Cloud Computing, Defining a Cloud, Benefits of Cloud Computing, Cloud Computing Architecture, Services Models (XaaS), Infrastructure as a Service, Platform as a Service, Software as a Service.

Unit II

Deployment Models, Public Cloud, Private Cloud, Hybrid Cloud, Community Cloud, Dynamic Provisioning and Resource Management, Virtualization: Characteristics of Virtualized Environment, Taxonomy of Virtualization Techniques, Pros and Cons of Virtualization, Xen, VMware, Hyper-V.

Unit III

Cloud Platform in Industry: Amazon Web Services- Compute Services, Storage Services, Communication Services, Additional Services, Google App Engine- Architecture and Core Concepts, Application Life Cycle, Cost Model, Microsoft Azure – Azure Core Concepts, SQL Azure, Windows Azure Platform Appliance.

Unit IV

Cloud Application: Scientific Applications- ECG Analysis in cloud, Protein Structure Prediction, Gene Expression data analysis for Cancer Diagnosis, Satellite Image Processing, Business and Consumer Applications-CRM and ERP, Productivity, Social Networking, Media Applications, Multiplayer Online gaming. Cloud Security.

Text and Reference Books:

1. Rajkumar Buyya, Christian Vecchiola and S ThamaraiSelvi, *Mastering Cloud Computing*, Tata McGraw Hill Education Pvt. Ltd., 2013.
2. Kai Hwang, Geofferyu C. Fox and Jack J. Dongarra, *Distributed and Cloud Computing*, Elsevier, 2012.
3. John W. Ritting and James F. Ransome, *Cloud Computing: Implementation Management and Security*, CRC press, 2012.

CO-PO Articulation Matrix Cloud Computing Course (PEC-CSE404-T)

List of Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1. **Define** concepts related to cloud computing. (LOTS: Level 1: Remember)	1	–	–	–	–	–	–	–	–	–	–	–	–	2	–
CO2. **Express** deployment models for clouds. (LOTS: Level 2: Understand)	1	–	–	–	–	–	–	–	–	–	–	–	–	3	–
CO3. **Apply** cloud computing techniques for various applications. (LOTS: Level 3: Apply)	2	2	2	–	2	–	–	–	–	–	–	–	–	3	–
CO4. **Analyse** cloud computing services used at various levels. (HOTS: Level 4: Analyse)	3	3	2	3	2	–	–	–	–	–	–	–	–	3	–
CO5. **Assess** real time cloud services. (HOTS: Level 5: Evaluate)	3	3	3	3	3	2	–	–	–	–	–	2	–	3	–
Level of Attainments PEC-CSE404-T															

Advanced Microprocessor

General Course Information

Course Code: PEC-CSE405-T/ PEC-IT405-T Course Credits: 3 Type: Professional Elective Contact Hours: 3hours/week Mode: Lectures (L) Examination Duration: 3 hours	**Course Assessment Methods (internal: 30; external: 70)** Two minor examinations (20 marks), Class Performance measured through percentage of lectures attended (4 marks), assignments (6 marks), and the end- semester examination (70 marks). For the end semester examination, nine questions are to be set by the examiner. A candidate is required to attempt 5 questions in all. All questions carry equal marks. Question number 1 will be compulsory and based on the entire syllabus. It will contain seven parts of 2 marks each. Question numbers 2 to 9 will be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt the remaining four questions by selecting one question from each of the four units..

Pre-requisites: Basic knowledge of Digital Electronics, Computer Architecture and Organization.

About the Course and its Outcomes:

A microprocessor incorporates the functions of a central processing unit (CPU) on a single integrated circuit. The advent of microprocessors and their increased capacity made them to be used in everything be it a smallest embedded system or handheld device, or the largest mainframe and supercomputer. It is being used in variety of applications such as process control systems, security systems, household appliances, and mobile phone technologies. This course aims to introduce the architecture, programming and interfacing of various hardware circuits with microprocessors. It would help the students learn the advanced techniques in the modern microprocessors and give them exposure to memory interfacing and management, monitoring and control applications, and the latest technologies.

Course outcomes: By the end of the course a student would be able to:

CO1. **describe** the features and use of the real and protected modes of microprocessors. (LOTS: Level 1: Remember)
CO2. **explain** the internal architecture of the 16, 32, and 64-bit microprocessors and compare and contrast the features of different Intel microprocessors. (LOTS: Level 2: Understand)
CO3. **analyse** memory, input/output and interrupt interfaces to the microprocessors. (HOTS: Level 4: Analyze)
CO4. **compare** the state-of-the-art technologies in the field of microprocessors.(HOTS: Level 5: Evaluate)
CO5. **design** the microprocessor based control systems and develop the software to control them. (HOTS: Level 6: Create)

Course Content

Unit I

Microprocessor 8086- Internal architecture, Real mode memory addressing, Protected mode memory addressing, Memory paging, Data addressing modes, Program memory addressing modes, Stack memory addressing modes, Directives and operators, Data transfer instructions, Arithmetic & logic instructions, Program control instructions, Data conversions, Assembly language programming.

Unit II

The Pin-Outs and pin functions of 8086 microprocessors and 8088 co-processor, Clock generator, Bus buffering and latching, Bus timings, READY and WAIT state, maximum mode and minimum mode configuration, Memory devices, Memory interface, Address decoding, 16 bit, 32 bit and 64 bit memory interface, I/O Programming, Programmed I/O, Interrupt I/O and DMA, I/O addresses and I/O ports .

Unit III

80286- features, Internal Architecture, bus interface, addressing modes; 80386-features, Internal Architecture, bus interface, addressing modes; 80486-features, Internal Architecture, bus interface, addressing modes; 16550 Programmable communications interface, Asynchronous serial data, Data acquisition system, Temperature monitoring system etc

Unit IV

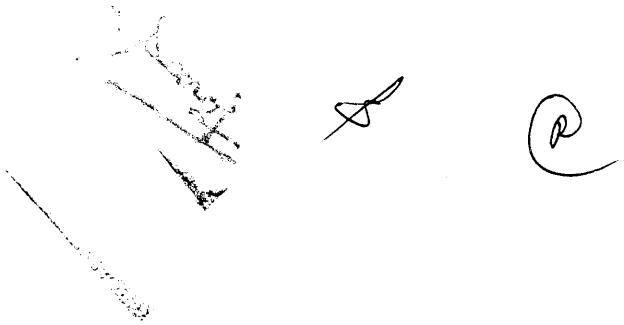
Pentium processor, The memory system, I/O system, Branch prediction logic, cache structure, superscalar architecture, special Pentium registers, Pentium memory management, Introduction to Pentium pro, Pentium II, Pentium III, Pentium IV and Core 2 microprocessors, Multi-core microprocessor architecture, Intel Hyper-Threading technology, Turbo Boost technology, state-of-the-art multi-core microprocessors.

Text and Reference Books:

1. Barry B. Brey, *INTEL Microprocessors*, 8th Edition, Prentice-Hall Inc., U.S.A., 2008.
2. Yu-cheng Liu, Glenn A. Gibson, *Microcomputer systems: The 8086 /8088 Family architecture, Programming and Design*, Second Edition, Prentice Hall of India, 2003
3. Walter A. Triebel, *The 80386, 80486, and Pentium Microprocessor: Hardware, Software, and Interfacing*, Prentice-Hall Inc., U.S.A., 1998.
4. K. Ray and K.M. Bhurchandi, *Intel Microprocessors: Architecture, Programming and Interfacing*, McGraw Hill Inc., 2001.
5. Shameem Akhter and Jason Roberts, *Multi-Core Programming*, Intel Press, 2006.
6. Douglas V. Hall, *Microprocessors and Interfacing: Programming and Hardware*, Tata McGraw-Hill, 1999.
7. James L. Antonakos , *The Pentium Microprocessor,* Pearson Education , 1997.

CO-PO Articulation Matrix Advanced Microprocessor Course (PEC-CSE405-T)

List of Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1. **Describe** the features and use of the real and protected modes of microprocessors. (LOTS: Level 1: Remember)	1	–	–	–	–	–	–	–	–	–	–	–	2	–	–
CO2. **Explain** and compare the internal architecture and features of the 16, 32, and 64-bit microprocessors (LOTS: Level 2: Understand)	1	–	–	–	–	–	–	–	–	–	–	–	3	–	–
CO3. **Demonstrate** the use of microprocessor related concepts and technologies for solving problems related to hardware design. (LOTS: Level 3: Apply)	2	3	2	–	–	–	–	–	–	–	–	–	3	–	–
CO4. **Analyse** memory, input/output and interrupt interfaces to the microprocessors. (HOTS: Level 4: Analyze)	2	3	2	2	2	–	–	–	–	–	–	–	3	–	–
CO5. **Compare and contrast** the state-of-the-art technologies in the field of microprocessors. (HOTS: Level 5: Evaluate)	3	3	2	2	2	–	–	–	–	–	–	–	3	–	–
CO6. **Design** the microprocessor-based control systems and develop the software to control them. (HOTS: Level 6: Create)	3	3	2	2	2	–	–	–	–	–	–	–	3	–	–
Level of Attainments PEC-CSE405-T															



Mobile Application Development

General Course Information

Course Code: PEC-CSE406-T/ PCC-IT403-T Course Credits: 3 Type: Professional Elective Contact Hours: 3 hours/week Mode: Lectures (L) Examination Duration: 3 hours	**Course Assessment Methods (internal: 30; external: 70)** Two minor examinations (20 marks), Class Performance measured through percentage of lectures attended (4 marks), assignments (6 marks), and the end- semester examination (70 marks). For the end semester examination, nine questions are to be set by the examiner. A candidate is required to attempt 5 questions in all. All questions carry equal marks. Question number 1 will be compulsory and based on the entire syllabus. It will contain seven parts of 2 marks each. Question numbers 2 to 9 will be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt the remaining four questions by selecting one question from each of the four units.

Pre-requisites: Java Programming and Object-Oriented programming, Knowledge of RDBMS and OLTP.

About the Course:

Mobile Application Development has been introduced as a Professional Elective course for Students of BTech(CSE/IT) keeping in view the Employers' requirements. Android Platform forms the basis for developing Mobile Applications since the last decade as compared to IOS Platform for Apple Products. The Environment requires User Interface to be developed using Buttons, Check-Boxes, Alert Dialog and its kind.

Course Outcomes: By the end of the course students will be able to:

CO1. **state** basic of Android , its Evolution and its Architecture. (LOTS: Level 1: Remember)
CO2. **demonstrate** the Lifecycle of Software for Android Mobile Applications. (LOTS: Level 2: Understand)
CO3. **prepare** Mobile Applications on the Android Platform. (LOTS: Level 3: Apply)
CO4. **compare** working with Buttons and other Widgets for Visual Environment. (HOTS: Level 4: Analyse)
CO5. **develop** Mobile Applications using data storage in SQLite Database and evaluate its Performance. (HOTS: Level 6: Create)

Course Content

Unit I

Mobile OS Architecture: Android, Blackberry OS, Firefox OS, IOS, Window OS, ARM and MIPS processor, Challenges of the mobile platform, Hello Android example, Internal Details, Dalvik VM, Software Stack, Android Core Building Blocks, Android Emulator, AndroidManifest.xml, R.java file, Hide Title Bar, Screen Orientation.

Unit II

UI Widgets: Working with Button, Toast, Custom Toast, Button, Toggle Button, Switch Button, Image Button, CheckBox, Alert Dialog, Spinner, AutoCompleteTextView, RatingBar, DatePicker, TimePicker, ProgressBar, Quick Contact Budge, Analog Clock and Digital Clock, Working with hardware Button, File Download.

Unit III

Activity, Intent & Fragment: Activity Lifecycle, Activity Example, Implicit Intent, Explicit Intent, Fragment Lifecycle, Fragment Example, Dynamic Fragment.

Android Menu: Option Menu, Context Menu, Popup Menu

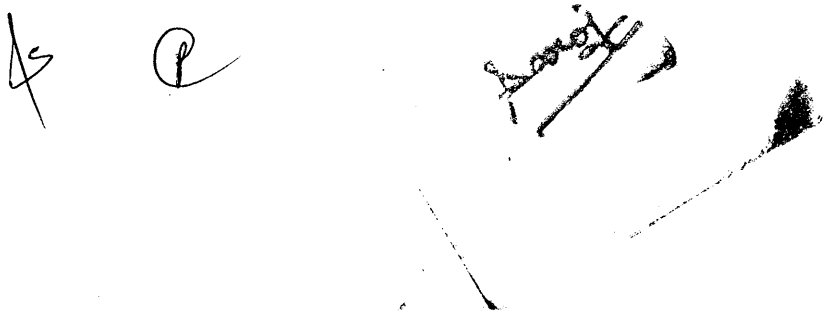
Layout Manager: Relative Layout, Linear Layout, Table Layout, Grid Layout.

Unit IV

Adaptor: Array Adaptor, ArrayList Adaptor, Base Adaptor.
View: GridView, WebView, ScrollView, SearchView, TabHost, DynamicListView, Expanded ListView.
SQLite: SQLite API, SQLite-Spinner, SQLite ListView
XML & JSON: XML Parsing SAX, XML Parsing DOM, XML Pull Parser, JSON basics, JSON Parsing.

Text and Reference Books:

1. Redazione Io Programmo, *Android Programming*, 2011
2. John,Horton, *Android Programming for Beginners*, packt publishing, 2015
3. Jason Wei, *Android Database Programming*, packt publishing, 2012
4. Mark L Murphy, *Android Programming Tutorials*, 3rd Edition, 2010
5. Bill Phillips et al., *Android Programming - The "Big Nerd Ranch"* Guide 2017
6. Rick Rogers et al., *Android Application Development: Programming with the Google SDK*, 2009



CO-PO Articulation Matrix Mobile Application Development Course (PEC-CSE406-T)

List of Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1. **State** basic of Android, its Evolution and its Architecture. (LOTS: Level 1: Remember)	1	–	–	–	–	–	–	–	–	–	–	–	3	–	–
CO2. **Demonstrate** the Lifecycle of Software for Android Mobile Applications. (LOTS: Level 2: Understand)	1	–	–	–	–	–	–	–	–	–	–	–	3	–	–
CO3. **Prepare** Mobile Applications on the Android Platform. (LOTS: Level 3: Apply)	2	2	2	–	2	–	–	–	–	–	–	–	3	2	–
CO4. **Compare** working with Buttons and other Widgets for Visual Environment. (HOTS: Level 4: Analyse)	–	–	2	–	1	–	–	–	–	–	–	–	3	–	–
CO5. **Develop** Mobile Applications using data storage in SQLite Database and evaluate its Performance. (HOTS: Level 6: Create)	3	2	2	3	3	3	–	–	2	–	1	2	3	3	–
Level of Attainments PEC-CSE406-T															

Multimedia Technologies

General Course Information

Course Code: PEC-CSE407-T / PEC-IT411-T Course Credits: 3 Type: Professional Elective Contact Hours: 3 hours/week Mode: Lectures (L) Examination Duration: 3 hours	**Course Assessment Methods (internal: 30; external: 70)** Two minor examinations (20 marks), Class Performance measured through percentage of lectures attended (4 marks), assignments (6 marks), and the end- semester examination (70 marks). For the end semester examination, nine questions are to be set by the examiner. A candidate is required to attempt 5 questions in all. All questions carry equal marks. Question number 1 will be compulsory and based on the entire syllabus. It will contain seven parts of 2 marks each. Question numbers 2 to 9 will be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt the remaining four questions by selecting one question from each of the four units.

Pre-requisites: Basics of Computer Graphics

About the Course:

Multimedia is a core and an essential course for every graduate in Computer Science and Engineering The objective of this course is to make students learn how to develop multimedia programs and demonstrate how still images, sound, and video can be digitized on the computer.

Course Outcomes: By the end of the course students will be able to:

CO1. **outline** the basic concepts of multimedia technology. (LOTS: Level 1: Remember)
CO2. **discuss the concepts** of animation, digitized sound, video control, and scanned images. (LOTS: Level 2: Understand)
CO3. **use** basic instructional design principles in the development of Multimedia. (LOTS: Level 3: Apply)
CO4. **compare** various audio and video file formats. (HOTS: Level 4: Analyse)
CO5. **devise** solutions for multimedia problems. (HOTS: Level 6: Create)

Course Content

Unit 1

Introduction to Multimedia concepts, Types of Multi-media Applications,

Methods to deliver Multimedia, Introduction to Multimedia Database, Multimedia Input and Output Devices.

Unit II

Introduction about font and faces, Using Text in Multimedia, Applying different types of text in multimedia

Font Editing and Design tools, Hypermedia and Hypertext application.

Unit III

The power of images, Making Still Images, Colouring, Image File Formats (GIF, JPEG, PNG etc.)

The power of sound, MIDI Vs. Digital Audio, Audio File Formats (AIFF, WAV, MPEG, MOV etc.)

Adding Sound to multimedia project.

Unit IV

Working of a Video and its Display, Digital Video Containers (Codecs & Video Format Converters)

Obtaining Video Clips, Shooting and editing Video, Non Linear Editing(NLE) in Videos

The stages of Multimedia Project, Hardware and Software requirements ,Authoring Systems

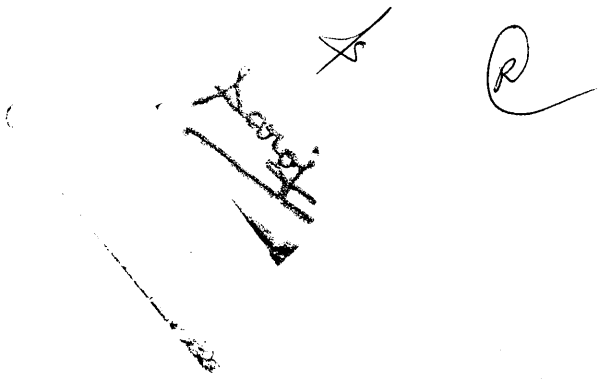
Team for Multimedia Development, Different stages of multimedia, The internet and multimedia

Text and Reference Books:

1. Tay Vaughan, *Multimedia: Making It Work*, Tata McGraw Hills, 2008.
2. James E Shuman, *Multimedia in Action*, Vikas Publishing House, 1997.
3. Andreas Holzinger, *Multimedia Basics Technology*, *Volume* 1, Firewall Media, 2005.
4. Rangan Parekh, *Principles of Multimedia*, Tata McGraw Hills, 2007.

CO-PO Articulation Matrix Multimedia Technologies Course (PEC-CSE407-T)

List of Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1. **Outline** the basic concepts of multimedia technology. (LOTS: Level 1: Remember)	1	–	–	–	–	–	–	–	–	–	–	–	2	–	–
CO2. **Discuss the concepts** of animation, digitized sound, video control, and scanned images. (LOTS: Level 2: Understand)	1	–	–	–	–	–	–	–	–	–	–	–	3	–	–
CO3. **Use** basic instructional design principles in the development of Multimedia. (LOTS: Level 3: Apply)	2	2	2	–	–	–	–	–	–	–	–	–	3	–	–
CO4. **Compare** various audio and video file formats. (HOTS: Level 4: Analyse)	2	2	2	–	2	–	–	–	–	–	–	–	3	–	–
CO5. **Devise** solutions for multimedia problems. (HOTS: Level 6: Create)	3	3	2	3	3	–	–	–	–	–	–	–	3	–	–
Level of Attainments PEC-CSE407-T															



Digital Image Processing

General Course Information

Course Code: PEC-CSE408-T/ PEC-IT408-T Course Credits: 3 Type: Professional Elective Contact Hours: 3hours/week Mode: Lectures (L) Examination Duration: 3 hours	**Course Assessment Methods (internal: 30; external: 70)** Two minor examinations (20 marks), Class Performance measured through percentage of lectures attended (4 marks), assignments (6 marks), and the end- semester examination (70 marks). For the end semester examination, nine questions are to be set by the examiner. A candidate is required to attempt 5 questions in all. All questions carry equal marks. Question number 1 will be compulsory and based on the entire syllabus. It will contain seven parts of 2 marks each. Question numbers 2 to 9 will be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt the remaining four questions by selecting one question from each of the four units.

Pre-requisites: knowledge of basic linear algebra, basic probability theory, basic programming techniques, and Fourier Transforms.

About the Course:

Digital Image Processing is a Professional Elective course that provides a theoretical foundation of digital image processing concepts. This course provides a mathematical foundation for digital manipulation of images, image acquisition, pre-processing, enhancement, segmentation and compression. Students learn algorithms that perform basic image processing operations (e.g., histogram processing, noise removal and image enhancement and restoration). Algorithms for image analysis (e.g., image compression, image segmentation and image representation) are explained.

Course Outcomes: By the end of the course students will be able to:

CO1. **state** concepts related to image acquisition and processing. (LOTS: Level 1: Remember)
CO2. **illustrate** the principles and methods in image processing. (LOTS: Level 2: Understand)
CO3. **apply** mathematical functions for digital manipulation of images such as image acquisition, pre-processing, segmentation, compression and representation. (LOTS: Level 3: Apply)
CO4. **compare** various image processing techniques. (HOTS: Level 4: Analyse)
CO5. **assess** the various image processing techniques for a given problem. (HOTS: Level 5: Evaluate)
CO6. **design** and implement algorithms for digital image processing operations such as histogram equalization, filtering, enhancement, restoration and denoising, segmentation, compression. (HOTS: Level 6: Create)

Course Content

Unit I

Introduction and fundamental to digital image processing: What is digital image processing, Origin of digital image processing, Examples that use digital image processing, Fundamental steps in digital image processing, Components of digital image processing system, Image sensing and acquisition, Image sampling, Quantization and representation, Basic relationship between pixels.Image enhancement in spatial domain and frequency domain: Background, Basic gray level transformation, Histogram processing, Basics of spatial filtering, Smoothing and sharpening spatial and the frequency domain filters.

Unit II

Image Restoration: Image degradation/restoration Process, Noise models, Restoration in presence of noise, Inverse filtering, Minimum mean square filtering, Geometric mean filter, Geometric transformations. Color Image Processing: Color fundamentals, Color models, Basics of full color image processing, Color transformations.

Unit III

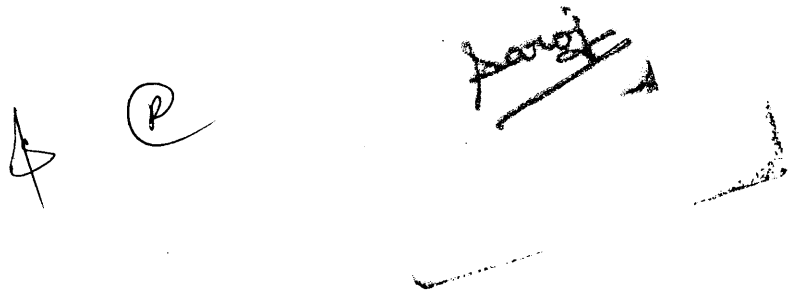
Image Compression: Fundamentals, Image compression models, Error free compression, Lossy compression. Image Segmentation: Detection of discontinuities, Edge linking and boundary detection, Thresholding, Region based segmentation.

Unit IV

Representation, Description and Recognition: Representation-chain codes, polygonal approximation and skeletons, Boundary descriptors-simple descriptors, shape numbers, Regional descriptors- simple, topological descriptors.
Recognition: Pattern and Pattern classes.

Text and Reference Books:

1. Rafael C. Gonzalez and Richard E. Woods, *Digital Image Processing*, Pearson Education, Ed, 2001.
2. Anil K. Jain, *Fundamentals of Digital Image Processing*, Pearson Education, PHI, 2001.
3. Tinku Acharya and Ajoy K. Ray, *Image Processing-Principles and Applications*, John Wiley & Sons,
4. Inc., 2005.
5. Chanda and D. Dutta Majumdar, *Digital Image Processing and Analysis*, PHI, 2003.
6. Milan Sonka, Vaclav Hlavac, Roger Boyle, *Image Processing, Analysis, and Machine Vision*, 2nd edition, PWS Publishing Company, Thomson Learning, 1999.



CO-PO Articulation Matrix Digital Image Processing Course (PEC-CSE408-T)

List of Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1. **State** concepts related to image acquisition and processing. (LOTS: Level 1: Remember)	1	3	–	–	–	–	–	–	–	–	–	–	–	–	2
CO2. **Illustrate** the principles and methods in image processing. (LOTS: Level 2: Understand)	1	–	–	–	–	–	–	–	–	–	–	–	–	–	3
CO3. **Apply** mathematical functions for digital manipulation of images such as image acquisition, pre-processing, segmentation, compression and representation. (LOTS: Level 3: Apply)	3	–	–	–	–	–	–	–	–	–	–	–	–	–	3
CO4. **Compare** various image processing techniques. (HOTS: Level 4: Analyse)	2	3	2	2	–	–	–	–	–	–	–	–	–	–	3
CO5. **Assess** the various image processing techniques for a given problem. (HOTS: Level 5: Evaluate)	3	3	2	2	–	–	–	–	–	–	–	–	–	–	3
CO6. **Design** and implement algorithms for digital image processing operations such as histogram equalization, filtering, enhancement, restoration and denoising, segmentation, compression. (HOTS: Level 6: Create)	3	3	2	3	3	–	–	–	–	–	–	2	–	–	3
Level of Attainments PEC-CSE408-T															

Advanced Microprocessor Lab.

General Course Information

Course Code: PEC-CSE405-P/ PEC-IT405-P Course Credits: 2 Type: Professional Elective Lab. Course Contact Hours: 3hours/week Mode: Lab. practice and assignments	**Course Assessment Methods (internal: 30; external: 70)** The internal and external assessment is based on the level of participation in lab. sessions and the timely submission of lab experiments/assignments, the quality of solutions designed for the assignments, the performance in VIVA-VOCE, the quality of lab. file and ethical practices followed. The internal examination is conducted by the course coordinator. The external examination is conducted by external examiner appointed by the Controller of Examination in association with the internal examiner appointed by the Chairperson of the Department.

Pre-requisites: Knowledge of assembly language.

About the Course:

The Lab work on Advanced Microprocessors cultivate the ability to write the programs by mastering the assembly language programming using various concepts like addressing modes, assemblers, directives, operators, interrupts. It makes students to get acquainted with the hardware specifications of various processors and operations between the microprocessor and input/output and/or memory devices. This Lab. fosters the ability to design microprocessors based applications.

Course outcomes: By the end of the lab course students will be able to:

CO1. **describe** the internal architecture of an X86 processor showing the general purpose registers, the segment registers, the ALU, the flags register, the instruction pointer (IP) register, and the instruction register. (LOTS: Level 2: Understand)

CO2. **implement** the assembly language programs for interfacing of peripherals/devices with processors. (HOTS: Level 6: Create)

CO3. **analyse** microprocessor controlled systems. (HOTS: Level 4: Analyse)

CO4. **evaluate** microprocessor controlled systems. (HOTS: Level 4: Analyse)

CO5. **create** Lab record for the assignments including aim, hardware and software requirements and solutions to given problems. (HOTS: Level 6: Create)

CO6. **demonstrate** independent enquiry, self-learning and ethical practices to solve unseen problems. (LOTS: Level 3: Apply).

List of experiments/assignments:

1. Three assignments on assembly language programs using 8086 Microprocessor.
2. Two assignments depicting the use of interrupts and interrupt structure.
3. Two/Three assignments based on addressing modes, operators and use of directives in assembly language programs.
4. Three assignments to show interfacing of 8086 with peripheral devices (I/O devices and memory).
5. Two assignments to design microprocessor-based applications such as rolling display.
6. Two assignments to program EEPROM chips to be used in applications such as traffic light controllers.
7. Two assignments based on Pentium multi-core microprocessors of 2.4 GHz/compatible bandwidth.

Note:

The actual experiments/assignments will be designed by the course coordinator. One assignment should be designed to be done in groups of two or three students. The assignments must meet the objective of the course and the levels of the given course outcomes. The list of assignments and schedule of submission will be prepared by the course coordinator at the beginning of the semester.

CO-PO Articulation Matrix Advanced Microprocessor Lab. Course (PEC-CSE405-P)

List of Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1. **Implement** the assembly language programs for interfacing of peripherals/devices with processors. (LOTS: Level 3: Apply)	2	–	–	–	–	–	–	–	–	–	–	–	2	–	–
CO2. **Describe** the internal architecture of an X86 processor showing the general purpose registers, the segment registers, the ALU, the flags register, the instruction pointer (IP) register, and the instruction register. (LOTS: Level 2: Understand)	1	–	–	–	2	–	–	–	–	–	–	–	3	–	–
CO3. **Analyse** Microprocessor controlled systems. (HOTS: Level 4: Analyse)	2	1	2	–	–	–	–	–	–	–	–	–	3	–	–
CO4. **Evaluate** Microprocessor controlled systems. (HOTS: Level 4: Analyse)	2	2	2	2	–	–	–	–	–	–	–	–	3	–	–
CO5. **Create** Lab record for the assignments including aim, hardware and software requirements and solutions to given problems. (HOTS: Level 6: Create)	–	–	–	–	–	–	–	–	–	3	–	–	–	–	–
CO6. **Demonstrate** independent enquiry, self-learning and ethical practices to solve unseen problems. (LOTS: Level 3: Apply).	–	–	–	–	–	–	–	3	3	–	–	3	–	–	–
Level of Attainments PEC-CSE405-P															

Mobile Application Development Lab.

General Course Information

Course Code: PEC-CSE406-P/ PCC-IT403-P Course Credits: 1 Type: Professional Elective Lab. Course Contact Hours: 2 hours/week Mode: Lab practice and assignments	**Course Assessment Methods (internal: 30; external: 70)** The internal and external assessment is based on the level of participation in lab. sessions and the timely submission of lab experiments/assignments, the quality of solutions designed for the assignments, the performance in VIVA-VOCE, the quality of lab. file and ethical practices followed. The internal examination is conducted by the course coordinator. The external examination is conducted by external examiner appointed by the Controller of Examination in association with the internal examiner appointed by the Chairperson of the Department.

Pre-requisites: Java programming, Object-oriented programming, RDBMS and OLTP

About the Course:

This course on Mobile Application Development is a developmental lab. work on Mobile programming. It incorporates creating Applications related to Android Studio framework. The objective of the lab course is to equip the students to solve the practical Mobile problems related to Application development.

Course Outcomes: By the end of the course students will be able to:

CO1. **apply** Android programming concepts for calling, display, creation and validation. (LOTS: Level 3: Apply)
CO2. **generate** solutions for content providers and permissive models. (HOTS: Level 6: Create)
CO3. **compare** the visual effects generated by Android and visual studio frameworks. (HOTS: Level 4: Analyse)
CO4. **design** applications for Android Programming by using Android Studio framework. (HOTS: Level 6: Create)
CO5. **create** lab record of the solutions for assignment. (HOTS: Level 6: Create)
CO6. **demonstrate** ethical practices, independent enquiry and self-learning to solve unseen problems. (LOTS: Level 3: Apply)

List of experiments/assignments:

1. Create "Hello World" application to display "Hello World" in the middle of the screen in red color with white background.
2. Create sample application with login module. (Check username and password), validate it for login screen or alert the user with a Toast.
3. Create and validate a login application using username as Email ID else login button must remain disabled.
4. Create a Login application and open a browser with any one search engine.
5. Create an application to display "Hello World" string the number of times user inputs a numeric value. (Example. If user enters 5, the next screen should print "Hello World" five times)

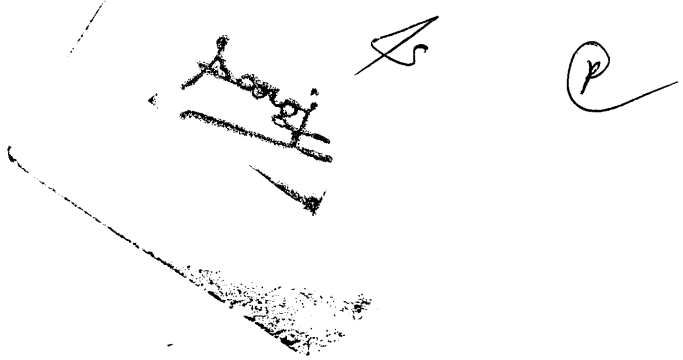
6. Create spinner with strings from the resource folder (res >> value folder). On changing spinner value, change image.
7. Create an application to change screen color as per the user choice from a menu.
8. Create a background application that will open activity on specific time.
9. Create an application that will have spinner with list of animation names. On selecting animation name, that animation should effect on the images displayed below.
10. Create an UI listing the engineering branches. If user selects a branch name, display the number of semesters and subjects in each semester.
11. Use content providers and permissions by implementing read phonebook contacts with content providers and display in the list.
12. Create an application to call a phone number entered by the user in the Edit Text box.
13. Create an application that will create database to store username and password.
14. Create an application to insert, update and delete a record from the database.

Note:

The actual experiments/assignments will be designed by the course coordinator. One assignment should be designed to be done in groups of two or three students. The assignments must meet the objective of the course and the levels of the given course outcomes. The list of assignments and schedule of submission will be prepared by the course coordinator at the beginning of the semester.

CO-PO Articulation Matrix Mobile Application Development Lab. Course (PEC-CSE406-P)

List of Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1. **Apply** Android programming concepts for calling, display, creation and validation. (LOTS: Level 3: Apply)	2	2	2	–	3	–	–	–	–	–	–	–	3	–	–
CO2. **Generate** solutions for content providers and permissive models. (HOTS: Level 6: Create)	2	2	2	2	3	–	–	–	–	–	–	–	3	–	–
CO3. **Compare** the visual effects generated by Android and visual studio frameworks. (HOTS: Level 4: Analyse)	2	2	2	2	3	–	–	–	–	–	–	–	3	–	–
CO4. **Design** applications for Android Programming by using Android Studio framework. (HOTS: Level 6: Create)	3	3	2	3	3	–	–	–	–	–	–	–	3	–	–
CO5. **Create** lab record of the solutions for assignment. (HOTS: Level 6: Create)	–	–	–	–	–	–	–	–	–	3	–	–	–	–	–
CO6. **Demonstrate** ethical practices, independent enquiry and self-learning to solve unseen problems. (LOTS: Level 3: Apply)	–	–	–	–	–	–	–	3	3	–	–	3	–	–	–
Level of Attainments PEC-CSE406-P															



Multimedia Technologies Lab.

General Course Information

Course Code: PEC-CSE407-P/ PEC-IT411-P Course Credits: 1 Type: Professional Elective Lab. Course Contact Hours: 2 hours/week Mode: Lab practice and assignments	**Course Assessment Methods (internal: 30; external: 70)** The internal and external assessment is based on the level of participation in lab. sessions and the timely submission of lab experiments/assignments, the quality of solutions designed for the assignments, the performance in VIVA-VOCE, the quality of lab. file and ethical practices followed. The internal examination is conducted by the course coordinator. The external examination is conducted by external examiner appointed by the Controller of Examination in association with the internal examiner appointed by the Chairperson of the Department.

Pre-requisites: Basic programming skills and knowledge of computer graphics.

About the Course:

This lab. course on Multimedia technologies involves a rigorous training on Adobe Photoshop, Macromedia Flash and blender. It incorporates solving problems related to animation and modelling framework. The objective of the lab course is to Learn to navigate and use modelling tools that will help students to gain a strong foundation in 3D design software Blender.

Course Outcomes: By the end of the course students will be able to:

CO1. **apply** the fundamental principles of different elements of multimedia. (LOTS: Level 3: Apply)
CO2. **use** modern tools for applying state-of-the art multimedia technologies. (LOTS: Level 3: Apply)
CO3. **analyse** various tools for an application. (HOTS: Level 4: Analyse)
CO4. **create** elegant posters, sceneries, animated stories and movie clips. (HOTS: Level 6: Create)
CO5. **creating** record of lab experiments. ((HOTS: Level 6: Create)
CO6. **demonstrate** ethical practices, self-learning and team work. (LOTS: Level 3: Apply)

List of experiments/assignments:

Adobe Photoshop

1. Introduction to Photoshop Basics.
2. Design a poster for 2019 elections and show the difference in quality and resolution for Print and Web.
3. Pick any picture of a magazine cover page and make changes using selection tool.
4. Draw a landscape using multiple Layers.
5. Paint a scenery of a park using different tools of Photoshop.
6. Take image from different Image Sources show variation in resolution.
7. Use effective cropping techniques to design a collage.
8. Design a scenery showing correction of image tonality.
9. Make a poster by adjusting Image Colours.
10. Painting the cover page of your magazine with Special Photoshop Tools.
11. Design a card on the occasion of Diwali using at least 3 different filters.
12. Make your passport size picture with all editing and print multiple copies of the same on A4 size page.

Macromedia Flash

13. Introduction to the layout and tools of Flash.
14. Move a car from left to right of the screen using symbols.
15. Design a movie clip.
16. Using timeline, design the casting of the movie directed by you.
17. Depict a small story using 2 D animation.

Blender

18. Introduction to Blender and its various tools.
19. Create an object using blender and show its motion.
20. Using Selections and Transform make a scenery.
21. Design a character for your game using modelling.
22. Depict the change in Materials, Lights and Rendering in 3 different frames.
23. Using Blender show compositing.

Note:

The actual experiments/assignments will be designed by the course coordinator. One assignment should be designed to be done in groups of two or three students. The assignments must meet the objective of the course and the levels of the given course outcomes. The list of assignments and schedule of submission will be prepared by the course coordinator at the beginning of the semester.

CO-PO Articulation Matrix Multimedia Technologies Lab. Course (PEC-CSE407-P)

List of Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1. **Apply** the fundamental principles of different elements of multimedia. (LOTS: Level 3: Apply)	2	1	–	–	–	–	–	–	–	–	–	–	2	–	–
CO2. **Use** modern tools for applying state-of-the art multimedia technologies. (LOTS: Level 3: Apply)	–	–	–	–	3	–	–	–	–	–	–	–	3	–	–
CO3. **Analyse** various tools for an application. (HOTS: Level 4: Analyse)	–	–	–	–	3	–	–	–	–	–	–	–	3	–	–
CO4. **Create** elegant posters, sceneries, animated stories and movie clips. (HOTS: Level 6: Create)	2	2	2	3		–	–	–	–	–	–	–	3	–	–
CO5. **Prepare** record of lab experiments. ((HOTS: Level 6: Create)	–	–	–	–	–	–	–	–	–	3	–	–	–	–	–
CO6. **Demonstrate** ethical practices, self-learning and team work. (LOTS: Level 3: Apply)	–	–	–	–	–	–	–	3	3	–	–	3	–	–	–
Level of Attainments PEC-CSE407-P															

Digital Image Processing Lab.

General Course Information

Course Code: PEC-CSE408-P/ PEC-IT408-P Course Credits: 1 Type: Professional Elective Lab. Course Contact Hours: 2 hours/week Mode: Lab practice and assignments	**Course Assessment Methods (internal: 30; external: 70)** The internal and external assessment is based on the level of participation in lab. sessions and the timely submission of lab experiments/assignments, the quality of solutions designed for the assignments, the performance in VIVA-VOCE, the quality of lab. file and ethical practices followed. The internal examination is conducted by the course coordinator. The external examination is conducted by external examiner appointed by the Controller of Examination in association with the internal examiner appointed by the Chairperson of the Department.

Pre-requisites: The students are expected to have a knowledge of computer graphics concepts.

About the Course:

This Lab course on Digital Image Processing is a developmental lab. work. It incorporates transformation of images in spatial and frequency domains, compression, restoration and reconstruction of images in SCILAB/MATLAB. The objective of the lab course is to equip the students to solve the practical Image processing problems.

Course Outcomes: By the end of the course students will be able to:

CO1. **implement** digital image processing concepts for image compression, restoration and reconstruction in SCILAB/MATLAB.(LOTS: Level 3: Apply)
CO2. **verify** the results of applying image processing problems to images (compression, expansion, multi-resolution processing etc.) (HOTS: Level 4: Analyze)
CO3. **measure** the quality of image after the digital image processing techniques are implemented to an image. (HOTS: Level 5: Evaluate)
CO4. **devise** solutions for Image Processing tasks problems. (HOTS: Level 6: Create)
CO5. **design** Lab record for the assignments including aim, hardware and software requirements and solutions to the given problems. (HOTS: Level 6: Create)
CO6. **use** ethical practices, independent enquiry, self-learning and team spirit. (LOTS: Level 3: Apply).

List of experiments/assignments

1. Two/Three introductory assignments on SCILAB/MATLAB.
2. Two assignments on Point processing and Pixel Operations e.g scan your signature and make it clean with thresholding.)
3. One/Two assignments on Image flipping.
4. Two assignments on Image Arithmetic such as Addition, subtraction, multiplication and division.
5. Create an application to display "Hello World" string the number of times user inputs a numeric value. (Example. If user enters 5, the next screen should print "Hello World" five times.)

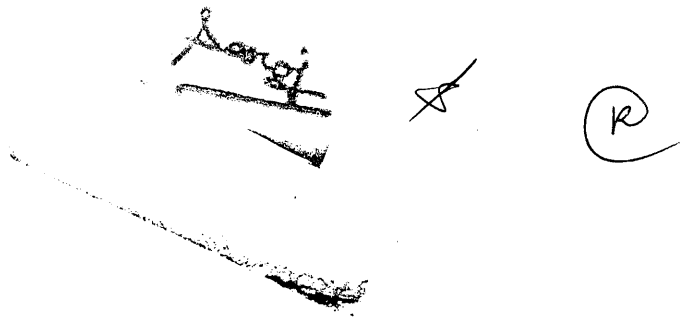
6. Two/Three assignments on performing Logical operations on Digital images such as NAND, NOR, EX-OR on these images.
7. Two/Three assignments on calculation and equalization of histogram for an input image.
8. Two/Three assignments on geometric transformation of image such as translation, Scaling, Rotation, Shrinking, Zooming.
9. One/Two assignments on adding noise to the image and apply image restoration techniques to improve quality of image.
10. Perform low pass and high pass filtering in frequency domain.

Note:

The actual experiments/assignments will be designed by the course coordinator. One assignment should be designed to be done in groups of two or three students. The assignments must meet the objective of the course and the levels of the given course outcomes. The list of assignments and schedule of submission will be prepared by the course coordinator at the beginning of the semester.

CO-PO Articulation Matrix Digital Image Processing Lab. Course (PEC-CSE408-P)

List of Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1. **Implement** digital image processing concepts for image compression, restoration and reconstruction in SCILAB/MATLAB. (LOTS: Level 3: Apply)	2	2	2	–	3	–	–	–	–	–	–	–	–	–	3
CO2. **Verify** the results of applying image processing problems to images (compression, expansion, multi-resolution processing etc.) (HOTS: Level 4: Analyze)	3	3	2	–	3	–	–	–	–	–	–	–	–	–	3
CO3. **Measure** the quality of image after the digital image processing techniques are implemented to an image. (HOTS: Level 5: Evaluate)	3	3	2	–	3	–	–	–	–	–	–	–	–	–	3
CO4. **Devise** solutions for Image Processing tasks problems. (HOTS: Level 6: Create)	3	3	3	3	3	–	–	–	–	–	–	–	–	–	3
CO5. **Design** Lab record for the assignments including aim, hardware and software requirements and solutions to the given problems. (HOTS: Level 6: Create)	–	–	–	–	–	–	–	–	–	3	–	–	–	–	–
CO6. **Use** ethical practices, independent enquiry, self-learning and team spirit. (LOTS: Level 3: Apply).	–	–	–	–	–	–	–	3	3	–	–	3	–	–	–
Level of Attainments PEC-CSE408-P															



Major Project I

General Course Information

Course Code: PROJ-CSE401 Course Credits: 4 Mode: Self learning under the guidance of faculty members. Contact hours: 8 hours/week	**Course Assessment Method (100)** The evaluation is done jointly by internal examiner and external examiner appointed by COE. The break-up of marks for evaluation is given below. 1. Literature review: 20 2. Problem formulation: 20 3. Basic knowledge of the tools: 20 4. Organisation and presentation of synopsis: 20 5. Level of Ethics followed: 20

About the major project I:

Students start working on their project work in seventh semester. Student do the background research for identifying appropriate problems, methodology and tools for their respective project works to be culminated in eighth semester. They prepare a synopsis of the project work to be carried out. At the end of seventh semester, each student is required to prepare a synopsis in the format provided and present it in front of a committee constituted by the Chairperson of the Department. Students can carry out projects in groups of two. In case of group project, the size of the problem should be significant, and members of the group must specify their individual contribution.

Course Outcomes: After doing Major Project Part 1 students will be able to:

CO1. **evaluate** critically the existing solutions and methodologies through reviewing literature. (HOTS: Level 5: Evaluate)
CO2. **formulate** suitable problems to be addressed. (HOTS: Level 6: Create)
CO3. **identify** tentative modern tools to solve the problem. (HOTS: Level 4: Analyse)
CO4. **organise** and communicate (written and oral) ideas effectively. (HOTS: Level 6: Create)
CO5. **develop** methodologies that meet ethical, societal and legal considerations. (HOTS: Level 6: Create)

CO-PO Articulation Matrix Major Project I (PROJ-CSE401)

List of Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1. **Evaluate** critically the existing solutions and methodologies through reviewing literature. (HOTS: Level 5: Evaluate)	2	3	3	3	–	–	–	–	–	–	–	3	–	–	–
CO2. **Formulate** suitable problems to be addressed. (HOTS: Level 6: Create)	2	3	3	3	–	–	–	–	–	–	–	–	–	–	–
CO3. **Identify** tentative modern tools to solve the problem. (HOTS: Level 4: Analyse)	2	–	2	–	3	–	–	–	–	–	–	2	–	–	–
CO4. **Organise** and communicate (written and oral) ideas effectively. (HOTS: Level 6: Create)	–	–	–	–	–	–	–	–	–	3	3	–	–	–	–
CO5. **Develop** methodologies that meet ethical, societal and legal considerations. (HOTS: Level 6: Create)	–	–	–	–	–	3	–	3	3	–	–	3	–	–	–
Level of Attainments PROJ-CSE401															

Mini Project using Open Source Tools /.NET

General Course Information

Course Code: PROJ-CSE402 *Course Credits: 1 Mode: Design and development of mini-project in lab. No. of hours per week: -	**Course Assessment Method (100)** Evaluation is done by internal examiner appointed by the Chairperson of the Department. The break-up of marks for evaluation is given below. 1. Significance and originality of the problem addressed and the solution provided: 20 2. Knowledge of the problem domain and the tool used (VIVA-VOCE):25 3. Report Writing: 20 4. Judgement of the open source tools learnt and quality of the solution developed: 20 5. Level of Ethics followed: 15

About the mini project:

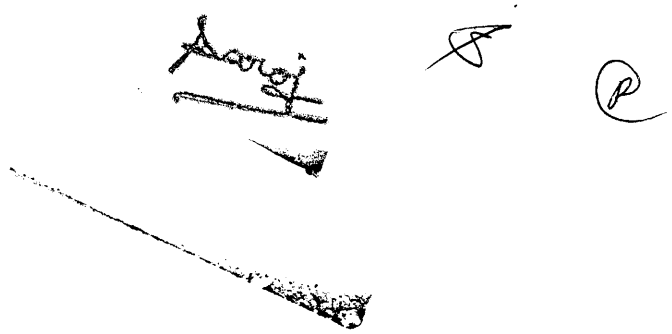
Students do a mini project using open source software after sixth semester. They are expected to learn any open source software and develop applications that can be completed within 4 to 6 weeks.

After doing mini-projects students will be able to

CO1. **identify** a suitable problem from the environment around. (HOTS: Level 4: Analyse)
CO2. **survey** the design of similar problems (HOTS: Level 5: Evaluate)
CO3. **select** suitable engineering specialisation and modern IT tools. (LOTS: Level 3: Apply)
CO4. **address** the problem in an original and innovative manner. (HOTS: Level 6: Create)
CO5. **communicate** orally as well as in written (mini project report) about the application developed. (HOTS: Level 6: Create)
CO6. **engage** in ethical practices, individual and team work, and lifelong learning. (LOTS: Level 3: Apply)

CO-PO Articulation Matrix Mini Project using Open Source Tools/.Net (PROJ-CSE402)

List of Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1. **Identify** a suitable problem from the environment around. (HOTS: Level 4: Analyse)	2	3	_	2	_	3	2	_	_	_	_	_	_	_	_
CO2. **Survey** the design of similar problems (HOTS: Level 5: Evaluate)	_	3	2	3	_	_	_	_	_	_	_	_	_	_	_
CO3. **Select** suitable engineering specialisation and modern IT tools. (LOTS: Level 3: Apply)	_	_	_	_	3	_	_	_	_	_	_	_	_	_	_
CO4. **Address** the problem in an original and innovative manner. (HOTS: Level 6: Create)	3	3	3	3	_	2	_	_	_	_	_	_	_	_	_
CO5. **Communicate** orally as well as in written (mini project report) about the application developed. (HOTS: Level 6: Create)	_	_	_	_	_	_	_	_	_	3	_	_	_	_	_
CO6. **Engage** in ethical practices and lifelong learning. (LOTS: Level 3: Apply)	_	_	_	_	_	_	_	3	3	_	_	3	_	_	_
Level of Attainments PROJ-CSE402															



Data Mining Techniques

General Course Information

Course Code: PCC-CSE403-T/ PCC-IT402-T Course Credits: 3 Type: Professional Core Contact Hours: 3 hours /week Mode: Lectures (L) Examination Duration: 3 hours	**Course Assessment Methods (internal: 30; external: 70)** Two minor examinations (20 marks), Class Performance measured through percentage of lectures attended (4 marks), assignments (6 marks), and the end- semester examination (70 marks). For the end semester examination, nine questions are to be set by the examiner. A candidate is required to attempt 5 questions in all. All questions carry equal marks. Question number 1 will be compulsory and based on the entire syllabus. It will contain seven parts of 2 marks each. Question numbers 2 to 9 will be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt the remaining four questions by selecting one question from each of the four units.

Pre-requisites: Knowledge of database systems, elementary knowledge of statistics and probability.

About the Course:

Today's era is the era of information. Data is growing exponentially day by day. There is a need to process and analyse the data to extract knowledge from it, so that one can use that knowledge for decision making. This course provides introductory concepts of data mining and data warehousing. The course will be taught with a database as well as machine learning perspectives. The objective of the course is to provide a comprehensive understanding of data prep-processing, data mining tasks and evaluation of results obtained out of data mining processes.

Course Outcomes: By the end of the course students will be able to:

CO1. **outline** various types of data mining and data warehouse concepts and techniques. (LOTS: Level 1: Remember)
CO2. **explain** characteristics, architecture of a data warehouse, OLAP operations and data mining tasks. (LOTS: Level 2: Understand)
CO3. **apply** various pre-processing and data mining techniques for extracting valuable information from data. (LOTS: Level 3: Apply)
CO4. **evaluate** the descriptive and predictive data mining models. (HOTS: Level 5: Evaluate)
CO5. **plan** a data mining process for discovering knowledge from real-world databases. (HOTS: Level 6: Create)

Course Content

Unit I

Introduction to Data Mining: Kind of data to be mined, Data Mining Functionalities, Technologies used in Data Mining, Applications of data Mining, Major Issues in Data Mining.

Data Pre-Processing: Need for preprocessing, Data Objects and Attribute types, Statistical description of data, Data Visualization, Measuring similarity and dissimilarity of data, Data Cleaning, Data Integration, Data Reduction, Data Transformation and Data Discretization.

Unit II

Data Warehouse: Introduction, Data Warehouse and Database Systems, Data Warehouse Architecture, Data Warehouse Models, Data Cube and OLAP, Multidimensional data Model, Concept Hierarchies, OLAP operations, Data Warehouse Implementation

Unit III

Mining Associations and Correlations: Mining Frequent Patterns, Associations and Correlations, Frequent Itemset Mining using Apriori Algorithm, Generating Association Rules from Frequent Itemsets. Improving efficiency of Apriori, Pattern Growth Approach for Mining Frequent Itemsets, Pattern evaluation Methods.

Advanced Pattern Mining: Pattern Mining in Multilevel and Multidimensional Space, Constraint-Based Frequent Pattern Mining.

Unit IV

Classification: Introduction, Classification using Decision Tree Induction, Bayesian Classification Methods, Rule Based Classification, Model Evaluation and Selection, Techniques to Improve Classification Accuracy. Classification by Backpropagation, Support Vector Machines and Lazy Learners.

Cluster Analysis: Introduction, Basic Clustering Methods, Partitioning Methods, Hierarchical Methods, Evaluation of Clustering.

Text and Reference Books:

1. Jiawei Han, Micheline Kamber and Jian Pei, *Data Mining Concepts and Techniques*, Morgan Kaufmann Publishers, Third Edition, July 2011.
2. Alex Berson, Stephen J. Smith, *Data Warehousing, Data Mining & OLAP*, Tata McGraw Hill, 2004.
3. Pang-Ning Tan, Michael Steinbach and Vipin Kumar, *Introduction to Data Mining*, Pearson Education, 2014.
4. K. P. Soman, Shyam Diwakar and V. Ajay, *Insight into Data Mining Theory and Practice*, Easter Economy Edition, Prentice Hall of India, 2009.
5. G. K. Gupta, *Introduction to Data Mining with Case Studies*, Prentice Hall of India, 2006.
6. Daniel T. Larose, *Data Mining Methods and Models*, Wiley, 2006.
7. W. H. Inman, *Building the Data Warehouse*, Wiley India, 2005.

CO-PO Articulation Matrix Data Mining Techniques (PCC-CSE403-T)

List of Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1. **Outline** various types of data mining and data warehouse concepts and techniques. (LOTS: Level 1: Remember)	1	–	–	–	–	–	–	–	–	–	–	–	–	–	2
CO2. **Explain** characteristics, architecture of a data warehouse, OLAP operations and data mining tasks. (LOTS: Level 2: Understand)	1	–	–	–	–	–	–	–	–	–	–	–	–	–	3
CO3. **Apply** various pre-processing and data mining techniques for extracting valuable information from data. (LOTS: Level 3: Apply)	2	1	–	–	2	–	–	–	–	–	–	–	–	–	3
CO4. **Evaluate** the descriptive and predictive data mining models. (HOTS: Level 5: Evaluate)	3	2	2	3	–	–	–	–	–	–	–	–	–	–	3
CO5. **Plan** a data mining process for discovering knowledge from real-world databases. (HOTS: Level 6: Create)	3	3	3	3	–	1	–	–	–	–	–	–	–	–	3
Level of Attainments PCC-CSE403-T															

Internet of Things

General Course Information

Course Code: PEC-CSE409-T/ PEC-CSE409-T Course Credits: 3 Type: Professional Elective Contact Hours: 3 hours/week Mode: Lectures (L) Examination Duration: 3 hours	**Course Assessment Methods (internal: 30; external: 70)** Two minor examinations (20 marks), Class Performance measured through percentage of lectures attended (4 marks), assignments (6 marks), and the end- semester examination (70 marks). For the end semester examination, nine questions are to be set by the examiner. A candidate is required to attempt 5 questions in all. All questions carry equal marks. Question number 1 will be compulsory and based on the entire syllabus. It will contain seven parts of 2 marks each. Question numbers 2 to 9 will be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt the remaining four questions by selecting one question from each of the four units.

Pre-requisites: Fundamentals of Computer Networks

About the Course:

The field of Internet of Things is growing very fast. The purpose of this course is to impart the knowledge on basic concepts of IoT, its Architecture, various protocols and applications in real world scenarios.

Course Outcomes: By the end of the course students will be able to:

CO1. **state** the basic concepts and key technologies of IoT. (LOTS: Level 1: Remember)
CO2. **discuss** the pros and cons of various protocols for IoT. (LOTS: Level 2: Understand)
CO3. **apply** the IOT models for business applications. (LOTS: Level 3: Apply)
CO4. **analyse** applications of IoT in real time scenario. (HOTS: Level 4: Analyse)
CO5. **design** business model scenarios (HOTS: Level 6: Create)

Course Content

Unit I

What is the Internet of Things? : History of IoT, About IoT, Overview and Motivations, Examples of Applications, Internet of Things Definitions and Frameworks : IoT Definitions, IoT Architecture, General Observations, ITU-T Views, Working Definition, IoT Frameworks, Basic Nodal Capabilities, Basics Of Microcontroller, Microprocessor Vs Microcontroller, Types of Sensor, Actuators and their Applications.

Unit II

Identification of IoT Objects and Services, Structural Aspects of the IoT, Environment Characteristics, Traffic Characteristics, Scalability, Interoperability, Security and Privacy, Open Architecture, Key IoT Technologies, Device Intelligence, Communication Capabilities, Mobility Support, Device Power, Sensor Technology, RFID Technology-Introduction, Principle of RFID, Components of an RFID system, Issues, Satellite Technology.

Unit III

IoT Access Technologies: Physical and MAC layers, Topology and Security of IEEE 802.15.4, 802.15.4g, 802.15.4e, 1901.2a, 802.11ah and LoRaWAN, Network Layer: IP versions, Constrained Nodes and Constrained Networks, Optimizing IP for IoT: From 6LoWPAN to 6Lo, Routing over Low Power and Lossy Networks, Application Transport Methods: Supervisory Control and Data Acquisition, Application Layer Protocols: CoAP and MQTT.

Unit IV

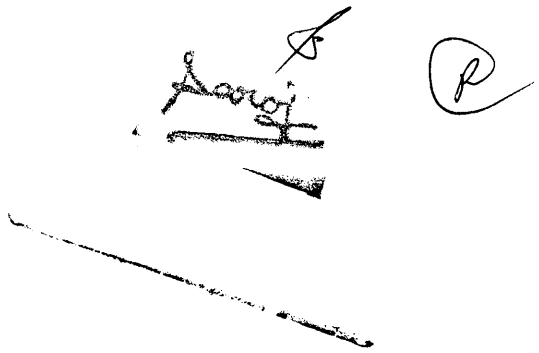
Business Models and Business Model Innovation, Value Creation in the Internet of Things, Business Model Scenarios for the Internet of Things. Internet of Things Applications: Smart Metering Advanced Metering Infrastructure, e-Health Body Area Networks, City Automation, Automotive Applications, Home Automation, Smart Cards, Smart Transportation and Smart Shopping.

Text and Reference Books:

1. David Hanes, Gonzalo Salgueiro, Patrick Grossetete, Rob Barton and Jerome Henry, *IoT Fundamentals: Networking Technologies, Protocols and Use Cases for Internet of Things*, Cisco Press, 1st Edition, 2017.
2. Olivier Hersent, David Boswarthick, Omar Elloumi , *The Internet of Things – Key applications and Protocols*, Wiley, 2nd Edition, 2012.
3. Dieter Uckelmann, Mark Harrison, Michahelles, Florian (Eds), *Architecting the Internet of Things*, 1st Edition, Springer, 2011.
4. Michael Margolis, Arduino Cookbook, "*Recipes to Begin, Expand, and Enhance Your Projects*", 2nd Edition, O'Reilly Media, 2011.
5. Arshdeep Bahga, Vijay Madisetti, *Internet of Things – A hands-on approach*, 1st Edition, Universities Press, 2015.

CO-PO Articulation Matrix Introduction to Internet of Things Course (PEC-CSE409-T)

List of Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1. **State** the basic concepts and key technologies of IoT. (LOTS: Level 1: Remember)	1	–	–	2	–	–	–	–	–	–	–	–	–	2	–
CO2. **Discuss** the pros and cons of various protocols for IoT. (LOTS: Level 2: Understand)	1	–	–	3	–	–	–	–	–	–	–	–	–	3	–
CO3. **Apply** the IOT models for business applications. (LOTS: Level 3: Apply)	2	2	2	3	3	–	–	–	–	–	–	–	2	3	–
CO4. **Analyse** applications of IoT in real time scenario. (HOTS: Level 4: Analyse)	3	3	2	–	3	–	–	–	–	–	–	–	2	3	2
CO5. **Design** business model scenarios (HOTS: Level 6: Create)	3	3	2	–	3	–	–	–	–	–	–	–	3	3	2
Level of Attainments PEC-CSE409-T															



Software Defined Networks

General Course Information

Course Code: PEC-CSE410-T/ PEC-IT410-T Course Credits: 3 Type: Professional Elective Contact Hours: 3hours/week Mode: Lectures (L) Examination Duration: 3 hours	**Course Assessment Methods (internal: 30; external: 70)** Two minor examinations (20 marks), Class Performance measured through percentage of lectures attended (4 marks), assignments (6 marks), and the end- semester examination (70 marks). For the end semester examination, nine questions are to be set by the examiner. A candidate is required to attempt 5 questions in all. All questions carry equal marks. Question number 1 will be compulsory and based on the entire syllabus. It will contain seven parts of 2 marks each. Question numbers 2 to 9 will be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt the remaining four questions by selecting one question from each of the four units.

Pre-requisites: Programming in C/C++/Java

About the Course:

Software Defined Networks is a result of improvement of flexibility of Network Control. To make the Networks Programmable it was deemed necessary to separate the Control Plane from the Data Plane. SDN Controllers are inserted into the Network to realize Network Virtualization. Openflow protocol and Mininet framework are used to design SDN. This Course is considered as a necessary addition in the Curriculum of B. Tech. (CSE/IT) from professional point of view.

Course Outcomes: By the end of the course students will be able to:

CO1. **outline** Software Defined Networks and its various components. (LOTS: Level 1: Remember)
CO2. **explain** techniques to make the Network Programmable for better flexibility. (LOTS: Level 2: Understand)
CO3. **use** of modern tools to implement SDN Controllers in a Network scenario. (LOTS: Level 3: Apply)
CO4. **breakdown** Virtual Networks into its components for controlling of networks. (HOTS: Level 4: Analyse)
CO5. **compare** and **contrast** the working of SDN through various protocols. (HOTS: Level 5: Evaluate)
CO6. **generate** SDN using Application Programming Interface and compute its performance for a given scenario. (HOTS: Level 6: Create)

Course Content

Unit I

Introduction: The need for Programmable Networks, Evolution of Software Defined Networks, Software Defined Networks' Architecture and Design, Traditional Switch Architecture, Centralized and decentralized Control Plane and Data Plane, IETF SDN framework, Scalability (Service provider Networks, ISP Automation), Reliability (QoS and Service Availability), Consistency (Configuration management and Access Control violations).

Unit II

Openflow and Software Defined Networks Controllers: Control and Data Plane Separation, Evolution of Openflow, SDN Controllers(POX, floodlight, openDayLight), Applicability of Openflow protocols in SDN Controllers, scalable Programming for SDN Controllers.

Unit III

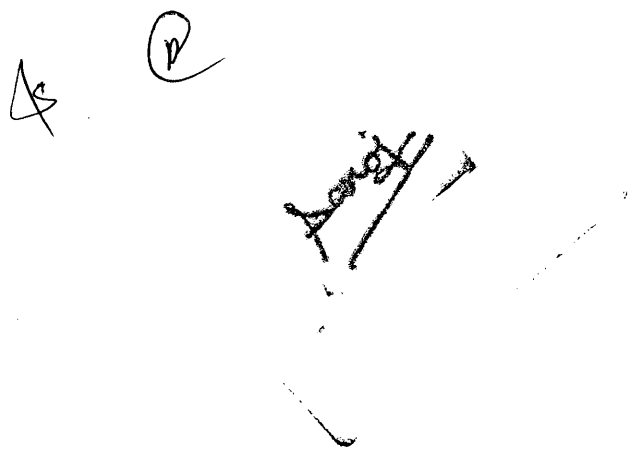
Network Virtualization: Virtual Network, Abstraction of physical Network, Components of Virtual Network (Virtual Switch, Bridge, Host-virtual adapter, NAT device, DHCP server, Network Adapter), Network as a Service (NaaS), Network Virtual Machine.

Unit IV

Software Defined Networks Programming: Programming Software Defined Networks, Northbound Application Programming Interface, Current Languages and tools, Network Functions Virtualization, Software Defined Networks implementation and Applications, Bandwidth Calendaring- Data Center Orchestration, Mininet. Use-cases(Network Access Control, Virtual Customer Edge, Data center Optimization), Latest trends in SDN.

Text and Reference Books:

1. Paul Goransson and Chuck Black, *Software Defined Networks: A Comprehensive Approach*, First Edition, Morgan Kaufmann, 2014.
2. Thomas D.Nadeau, Ken Gray, *Software Defined Networks*, O'Reilly Media, 2013.
3. Siamak Azodolmolky, *Software Defined Networking with Openflow, Packt Publishing*, 2013.
4. Kingston Smiler, *Openflow Cookbook*, Packt Publishing, 2015.
5. Doug Marschke, Jeff Doyle, PeteMoyer, *Software Defined Networking: Anatomy of Openflow*, Volume-I, Lulu Publishing Services, 2015.



CO-PO Articulation Matrix Software Defined Networks Course (PEC-CSE410-T)

List of Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1. **Outline** Software Defined Networks and its various components. (LOTS: Level 1: Remember)	1	–	–	–	–	–	–	–	–	–	–	–	–	2	–
CO2. **Explain** techniques to make the Network Programmable for better flexibility. (LOTS: Level 2: Understand)	1	–	–	–	–	–	–	–	–	–	–	–	–	3	–
CO3. **Use** of modern tools to implement SDN Controllers in a Network scenario. (LOTS: Level 3: Apply)	2	2	2	2	3	–	–	–	–	–	–	–	–	3	–
CO4. **Breakdown** Virtual Networks into its components for controlling of networks. (HOTS: Level 4: Analyse)	3	2	2	3	3	–	–	–	–	–	–	–	–	3	–
CO5. **Compare** and **contrast** the working of SDN through various protocols. (HOTS: Level 5: Evaluate)	3	3	2	3	3	–	–	–	–	–	–	–	–	3	–
CO6. **Generate** SDN using Application Programming Interface and compute its performance for a given scenario. (HOTS: Level 6: Create)	3	3	2	2	3	–	–	–	–	–	–	–	–	3	–
Level of Attainments PEC-CSE-410-T															

Network Administration and Management

General Course Information

Course Code: PEC-CSE411-T/ PCC-IT305-T Course Credits: 3 Type: Professional Elective Contact Hours: 3 hours/week Mode: Lectures (L) Examination Duration: 3 hours	**Course Assessment Methods (internal: 30; external: 70)** Two minor examinations (20 marks), Class Performance measured through percentage of lectures attended (4 marks), assignments (6 marks), and the end- semester examination (70 marks). For the end semester examination, nine questions are to be set by the examiner. A candidate is required to attempt 5 questions in all. All questions carry equal marks. Question number 1 will be compulsory and based on the entire syllabus. It will contain seven parts of 2 marks each. Question numbers 2 to 9 will be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt the remaining four questions by selecting one question from each of the four units.

Pre-requisites: Networking, protocols defined in layered Architecture, programming fundamentals.

About the Course:

Network Administration and Management is a Professional Elective course deemed to be necessary during the present era of Information Technology and Computer Science. This course deals with analyzing Network for statistics such as protocols, servers, memory, CPU etc. Network Monitoring and Management deals with different events in various types of platforms for response.

Course Outcomes: By the end of the course students will be able to:

CO1. **define** Network Administration and its various components. (LOTS: Level 1: Remember)
CO2. **distinguish** Network Administration and its Management on various platforms. (LOTS: Level 2: Understand)
CO3. **classify** the output for different responses to events by interpreting Network Monitoring statistics. (LOTS: Level 3: Apply)
CO4. **separate** portions of Network for troubleshooting using various tools. (HOTS: Level 4: Analyse)
CO5. **combine** Network Administration, Network Management and Network Monitoring into a one scenario and compute the performance of the integrated environment. (HOTS: Level 6: Create)

Course Content

Unit I

Network Administration: Introduction to Network Administration Approaches, Addressing, Subnetting and Supernetting, Fixed Vs Variable Masks, VLAN Principles and Configuration, Routing Concepts: Static and Dynamic Routing, Routing Protocols: RIP, OSPF, BGP. Network Address Translation (NAT), Configuring a Windows Box as a Router, Dial-up configuration and Authentication: PPP, Radius, RAS. Configuring a DNS Server in windows, Configuring Sendmail Service, Configuring a Web Server, Configuring a Proxy Server, TCP/IP Troubleshooting: ping, traceroute, ifconfig, netstat, ipconfig.

Unit II

Linux Network Administration: Setting up a file server, setting up samba server, configuring Network services: installing and configuring DHCP server, installing and configuring DNS server, setting up internal NTP server, hosting http content via Apache, sharing resources in a Network.

Unit III

Network management: Management Standards and models, Configuration Management and auto discovery, Fault Management, Fault identification and isolation, Event correlation techniques, SNMPv1, SNMPv2: Structure of Management Information, Standard Management Information Base (MIBs), MIB-II, Network Management Functions: Accounting Management, Performance Management, Network Usage, Metrics, and Quotas, SNMPv3: Protocol, MIB.

Unit IV

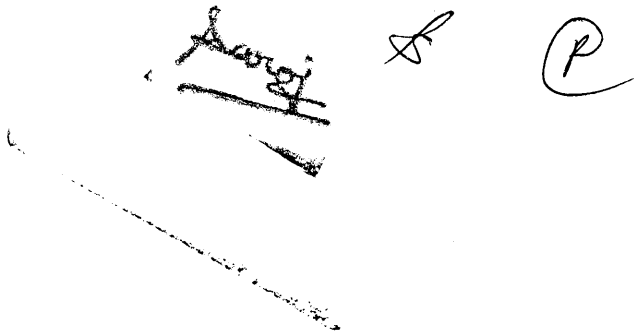
Network Monitoring: Network Performance Monitoring, Remote Network Monitoring (RMON1): Statistics Collection, Alarms and Filters, RMON2: Monitoring Network Protocol Traffic, Application-Layer Visibility, Management Tools, Systems and Applications: Test and Monitoring tools, Integrating tools, Development tools, Web-based Enterprise Management.

Text and Reference Books:

1. Mark Burgess, *Principles of Network and System Administration*, 2nd Edition, Wiley publications, 2004.
2. Craig Hunt, *TCP/IP Network Administration*, 3rd Edition, O'Reilly Publications, 2002.
3. George Splading, *Windows 2000 Administration*, Tata McGraw-Hill, 2000.
4. Tony Bautts, Terry Dawson, and Gregor N. Purdy, *Linux Network Administrator's Guide*, 3rd Edition, O'Reilly publications, 2005.

CO-PO Articulation Matrix Network Administration and Management Course (PEC-CSE411-T)

List of Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1. **Define** Network Administration and its various components. (LOTS: Level 1: Remember)	1	–	–	–	–	–	–	–	–	–	–	–	–	3	–
CO2. **Distinguish** Network Administration and its Management on various platforms. (LOTS: Level 2: Understand)	1	–	–	–	–	–	–	–	–	–	–	–	–	3	–
CO3. **Classify** the output for different responses to events by interpreting Network Monitoring statistics. (LOTS: Level 3: Apply)	2	2	2	3	3	–	–	–	–	–	–	–	–	3	–
CO4. **Separate** portions of Network for troubleshooting using various tools. (HOTS: Level 4: Analyse)	2	3	2	2	3	–	–	–	–	–	–	–	–	3	–
CO5. **Combine** Network Administration, Network Management and Network Monitoring into a one scenario and compute the performance of the integrated environment. (HOTS: Level 6: Create)	3	3	2	3	3	–	–	–	–	–	–	–	–	3	–
Level of Attainments : PEC-CSE-411-T															



Software Testing and Quality Assurance

General Course Information

Course Code: PEC-CSE412-T/ PEC-IT412-T Course Credits: 3 Type: Professional Elective Contact Hours: 3 hours/week Mode: Lectures (L) Examination Duration: 3 hours	**Course Assessment Methods (internal: 30; external: 70)** Two minor examinations (20 marks), Class Performance measured through percentage of lectures attended (4 marks), assignments (6 marks), and the end- semester examination (70 marks). For the end semester examination, nine questions are to be set by the examiner. A candidate is required to attempt 5 questions in all. All questions carry equal marks. Question number 1 will be compulsory and based on the entire syllabus. It will contain seven parts of 2 marks each. Question numbers 2 to 9 will be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt the remaining four questions by selecting one question from each of the four units.

Pre-requisites: Software Engineering.

About the Course:

This course introduces students to software testing process and describes the quality assurance process and its role in software development. During the course students learns about the testing methods and tools, creating good test cases to improve the quality of software.

Course Outcomes: By the end of the course students will be able to:

CO1. **recall** the process of software testing life cycle and quality assurance. (LOTS: Level 1: Remember)
CO2. **demonstrate** reusability testing on software applications. (LOTS: Level 2: Understand))
CO3. **apply** software testing tools for predicting the behavior of software applications. (LOTS: Level 3: Apply)
CO4. **identify** the test cases for software applications. (HOTS: Level 4: Analyse)
CO5. **plan** test cases and quality management activities. (HOTS: Level 6: Create)
CO6. **predict** software quality based on quality parameters and quality models. (HOTS: Level 6: Create)

Course Content

Unit I

Introduction to Basic of software testing & Terminology, Software Development & Software Testing Life Cycle- role and activities, Necessity and Objectives of testing; Quality Concepts, Quality Control, McCall's factor model; Different Software Development Model; Object– oriented testing, Web testing, GUI testing; Elements of Software quality assurance; Quality Assurance Activities, Statistical Quality Assurance; Software Reliability, SQA plan , Quality Standards:-IEEE, CMM, ANSI.

Unit II

Testing Concepts, Issues and Techniques, Levels of Testing, Verification and Validation Model ; Techniques of Verification:-Peer Review, Walkthrough, Inspection, FTR ; Unit testing, Integration testing, Function Testing ; System testing, Installation Testing, Usability Testing, Regression testing, ; Performance testing:-Load Testing, Stress Testing, Security testing, Volume testing ; Acceptance testing:-Alpha testing, Beta testing, Gamma testing.

Unit III

Black Box Testing Methods: Equivalence partitioning, Boundary-value analysis, Error guessing, graph- based testing methods, Decision Table Testing; White Box Testing Methods: Statement coverage, Decision coverage, Condition coverage, Path testing, Data flow testing.

Test Planning & Documentation: Development plan and quality plan objectives; Testing Strategy, Test Management, Strategic Management, Operational Test Management, Managing the Test Team, Test Plans, Test Cases, Test Data, Risk Analysis.

Unit IV

Testing Tools, Features of test tool; Guidelines for selecting a tool; Tools and skills of tester; Static testing tools, Dynamic testing tools, Advantages and disadvantages of using tools, Introduction to open source testing tool.

Text and reference books:

1. M. G. Limaye, *Software Testing Principles, Techniques and Tools*, TMH, 2009.
2. Yogesh Singh, *Software Testing*, Cambridge University Press, 2016.
3. Ron Pattorn, *Software Testing*, 2nd edition, Sams, 2005.
4. Roger S. Pressman, Software Engineering- a Practitioners approach, 8th edition, McGraw Hill, 2014
5. Jeff Tian, *Software Quality Engineering: Testing, Quality Assurance and Quantifiable Improvement*, Wiley, 2005.
6. Stephan H. Kan, *Metrics and Models in Software Quality Engineering*, 2nd edition, Addison-Wesley, 2009.
7. William E. Perry, *Effective Methods of Software Testing*, 2nd edition, Wiley, 2000.

CO-PO Articulation Matrix Software Testing and Quality Assurance Course (PEC-CSE412-T)

List of Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1. **Recall** the process of software testing life cycle and quality assurance. (LOTS: Level 1: Remember)	1	–	–	–	–	–	–	–	–	–	–	–	3	–	–
CO2. **Demonstrate** reusability testing on software applications. (LOTS: Level 2: Understand))	1	–	–	–	–	–	–	–	–	–	–	–	3	–	–
CO3. **Apply** software testing tools for predicting the behavior of software applications. (LOTS: Level 3: Apply)	2	2	2	2	3	–	–	–	–	–	–	–	3	–	–
CO4. **Identify** the test cases for software applications. (HOTS: Level 4: Analyse)	2	3	2	3	–	–	–	–	–	–	–	–	3	–	–
CO5. **Plan** test cases and quality management activities. (HOTS: Level 6: Create)	3	3	3	3	3	–	–	–	–	–	–	–	3	–	–
CO6. **Predict** software quality based on quality parameters and quality models. (HOTS: Level 6: Create)	3	3	2	3	3	–	–	–	–	–	–	–	3	–	–
Level of Attainments PEC-CSE412-T															

Machine Learning

General Course Information

Course Code: PEC-CSE413-T/ PEC-IT413-T Course Credits: 3 Type: Professional Elective Contact Hours: 3 Mode: Lectures (L) Examination Duration: 3 hours	**Course Assessment Methods (internal: 30; external: 70)** Two minor examinations (20 marks), Class Performance measured through percentage of lectures attended (4 marks), assignments (6 marks), and the end- semester examination (70 marks). For the end semester examination, nine questions are to be set by the examiner. A candidate is required to attempt 5 questions in all. All questions carry equal marks. Question number 1 will be compulsory and based on the entire syllabus. It will contain seven parts of 2 marks each. Question numbers 2 to 9 will be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt the remaining four questions by selecting one question from each of the four units.

Pre-requisites: Basics of Linear Algebra and Statistics, Basics of Probability Theory, Data Structures and Computer Algorithms.

About the Course:

Machine learning is the study of computer algorithms that improve their performance through experience. Machine learning draws its conceptual foundation from the fields like artificial intelligence, probability and statistics, computational complexity, cognitive science, biology and information theory etc. The course introduces some of the key machine learning algorithms and the theory that form the backbone of these algorithms. The examples of such algorithms are classification algorithms for learning patterns from data, clustering algorithms for grouping objects based on similarity, neural network algorithms for pattern recognition, genetic algorithms for searching large and complex search spaces etc.

Course Outcomes: By the end of the course students will be able to:

CO1. **outline** the concepts and working of different machine learning algorithms. (LOTS: Level 1: Remember)
CO2. **interpret** the results of machine learning algorithms. (LOTS: Level 2: Understand)
CO3. **apply** machine learning concepts and algorithms to given problems. (LOTS: Level 3: Apply)
CO4. **analyse** the performance of machine learning algorithms. ((HOTS: Level 4: Analyse)
CO5. **compare and contrast** different machine learning algorithms. (HOTS: Level 5: Evaluate)
CO6. **design** machine learning algorithms for optimization, pattern recognition and search problems. (HOTS: Level 6: Create)

Course Content

Unit I

Introduction: Well posed learning problems, designing a learning system, Issues in machine learning, the concept learning task, Concept learning as search, Finding a maximally specific hypothesis, Version spaces and candidate elimination algorithm, Remarks on version spaces and candidate-eliminations, Inductive bias.

Unit II

Supervised Learning: Introduction to linear regression, estimating the coefficients, Accessing the accuracy of the coefficient estimates, Accessing the accuracy of the regression model, Multiple linear regression, Logistic regression, basic decision tree learning (ID3) algorithm, Hypothesis space search in decision tree learning algorithm, Inductive bias in decision tree learning, Issues in decision tree learning, k-nearest neighbour learning.

Unit III

Unsupervised Learning: About clustering, type of data in clustering analysis, k-means and k-medoids, DBSCAN density-based clustering method, Performance analysis of clustering algorithms,

Artificial Neural networks: Neural Network representations, Appropriate problems for neural network learning, Perceptron. The perceptron training rule, Gradient descent and delta rule, Multilayer Networks and back propagation algorithm.

Unit IV

Bayesian Learning: Bayes theorem, Bayes theorem and concept learning, Maximum likelihood and least-squared error hypotheses, Naïve Bayes Classifier.

Evaluating Hypotheses: Estimating hypothesis Accuracy, Basics of sampling theory, Error estimation and estimating Binomial proportions, the binomial distribution, Mean and variance, Bias and variance, Confidence intervals, Two sided or one sided bounds, Central limit theorem, Hypothesis testing, Comparing learning algorithms

Text and Reference Books:

1. Tom M. *Mitchell, Machine Learning*, McGraw-Hill, 1997.
2. Bishop Christopher, *Pattern Recognition and Machine Learning*, Springer Verlag, 2006.
3. Trevor Hastie, Robert Tibshirani, Jerome Friedman, *The Elements of Statistical Learning: Data Mining, Inference and Prediction*, Springer, 2nd edition, 2009..J. Han and M. Kamber, Data Mining Concepts and Techniques, 3rd Edition, Elsevier, 2012.
4. S. Rajeshkaran, G. A. Vijayalakshmi Pai, *Neural Networks, Fuzzy Logic and Genetic Algorithm: Synthesis and Applications*, PHI, 2003.

CO-PO Articulation Matrix Machine Learning Course (PEC-CSE413-T)

List of Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1. **Outline** the concepts and working of different machine learning algorithms. (LOTS: Level 1: Remember)	1	–	–	1	–	–	–	–	–	–	–	–	–	–	3
CO2. **Interpret** the results of machine learning algorithms. (LOTS: Level 2: Understand)	2	2	2	3	–	–	–	–	–	–	–	–	–	–	3
CO3. **Apply** machine learning concepts and algorithms to given problems. (LOTS: Level 3: Apply)	2	2	2	2	–	–	–	–	–	–	–	–	–	–	3
CO4. **Analyse** the performance of machine learning algorithms. ((HOTS: Level 4: Analyse)	3	3	2	3	–	–	–	–	–	–	–	–	–	–	3
CO5. **Compare and contrast** different machine learning algorithms. (HOTS: Level 5: Evaluate)	3	3	2	3	–	–	–	–	–	–	–	–	–	–	3
CO6. **Design** machine learning algorithms for optimization, pattern recognition and search problems. (HOTS: Level 6: Create)	3	3	2	3	–	–	–	–	–	–	–	–	–	–	3
Level of Attainments PEC-CSE413-T															

Big Data Analytics

General Course Information

Course Code: PEC-CSE414-T/ PEC-IT414-T Course Credits: 3 Type: Professional Elective Contact Hours: 3hours/week Mode: Lectures (L) Examination Duration: 3 hours	**Course Assessment Methods (internal: 30; external: 70)** Two minor examinations (20 marks), Class Performance measured through percentage of lectures attended (4 marks), assignments (6 marks), and the end- semester examination (70 marks). For the end semester examination, nine questions are to be set by the examiner. A candidate is required to attempt 5 questions in all. All questions carry equal marks. Question number 1 will be compulsory and based on the entire syllabus. It will contain seven parts of 2 marks each. Question numbers 2 to 9 will be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt the remaining four questions by selecting one question from each of the four units.

Pre-requisites: Basics of statistics and data mining.

About the Course:

This course aims to provide students with the knowledge of current challenges, methodologies and technologies in processing big data. Emphasis will be placed on the students' understanding of the rationales behind the technologies and the students' ability to analyse big data using professional packages and tools.

Course Outcomes: By the end of the course students will be able to:

CO1. **recall** the concepts of big data analysis. (LOTS: Level 1: Remember)
CO2. **interpret** the outcomes of big data analysis. (LOTS: Level 2: Understand)
CO3. **apply** technical skills and modern tools for descriptive and predicative modelling. (LOTS: Level 3: Apply)
CO4. **analyse** a framework for visualization of big data analytics for business user. (HOTS: Level 4: Analyse)
CO5. **examine** critically the results of mining to support business decision-making. (HOTS: Level 5: Evaluate)
CO6. **design** schemes for big data analytics for solving big data problems in efficient manner. (HOTS: Level 6: Create)

Course Content

Unit I

Introduction: Overviews of Big Data, State of the Practice in Analytics, The Data Scientist, Big Data Analytics in Industry Verticals, Data Analytics Lifecycle Challenges of Conventional Systems, Statistical Concepts: Sampling Distributions, Re-Sampling, Statistical Inference, Prediction Error, Regression Modelling, Multivariate Analysis, Bayesian Modelling.

Unit II

Mining Data Streams: Stream Data Model and Architecture, Stream Computing, Sampling Data in a Stream, Filtering Streams, Counting Distinct Elements in a Stream, Estimating Moments, Counting Oneness in a Window, Decaying Window, Real time Analytics, Platform (RTAP) Applications, Case Studies, Real Time Sentiment Analysis, Stock Market Prediction

Unit III

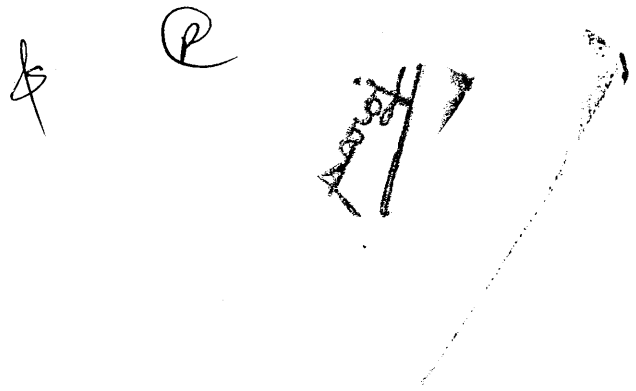
Frequent Itemset and Clustering: Mining Frequent Itemsets, Market Based Model: Apriori Algorithm, Handling Large Data Sets in Main Memory, Limited Pass Algorithm, Counting Frequent Itemsets in a Stream, Clustering based Techniques: Hierarchical, K-Means etc., Clustering High Dimensional Data, CLIQUE And PROCLUS, Frequent Pattern based Clustering Methods, Clustering in Non-Euclidean Space, Clustering for Streams and Parallelism..

Unit IV

Frameworks and Visualization: Overview of MapReduce, Hadoop, Hive, MapR, Sharding, NoSQL Databases, S3, HADOOP, Distributed File System (HDFS), Visualizations: Visual Data Analysis Techniques, Interaction Technique and Applications.

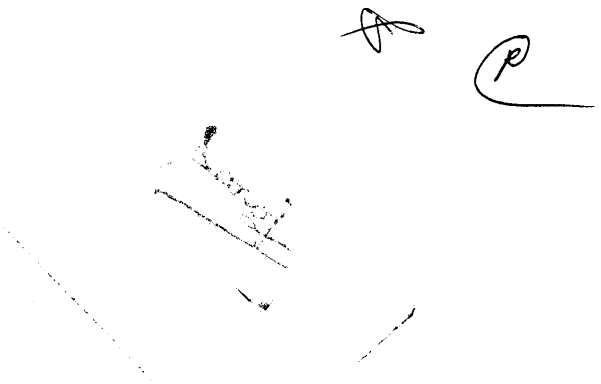
Text and Reference Books:

1. Michael Berthold, David J. Hand, *Intelligent Data Analysis*, Springer, 2007.
2. A. Rajaraman, J.D. Ullman, *Mining of Massive Datasets*, Cambridge University Press, 2012.
3. Bill Franks, *Taming the Big Data Tidal Wave: Finding Opportunities in Huge Data Streams with Advanced Analytics*, John Wiley & sons, 2012.
4. Glenn J. Myatt, *Making Sense of Data*, John Wiley & Sons, 2007
5. Pete Warden, *Big Data Glossary*, O'Reilly, 2011.



CO-PO Articulation Matrix Big Data Analytics Course (PEC-CSE414-T)

List of Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1. **Recall** the concepts of big data analysis. (LOTS: Level 1: Remember)	1	1	–	1	–	–	–	–	–	–	–	–	–	–	2
CO2. **Interpret** the outcomes of big data analysis. (LOTS: Level 2: Understand)	2	2	2	2	–	–	–	–	–	–	–	–	–	–	3
CO3. **Apply** technical skills and modern tools for descriptive and predicative modelling. (LOTS: Level 3: Apply)	3	2	2	2	–	–	–	–	–	–	–	–	–	–	3
CO4. **Analyse** a framework for visualization of big data analytics for business user. (HOTS: Level 4: Analyse)	3	2	2	3	–	–	–	–	–	–	–	–	–	–	3
CO5. **Examine** critically the results of mining to support business decision-making. (HOTS: Level 5: Evaluate)	3	2	2	3	–	–	–	–	–	–	–	–	–	–	3
CO6. **Design** schemes for big data analytics for solving big data problems in efficient manner. (HOTS: Level 6: Create)	3	2	2	3	–	–	–	–	–	–	–	–	–	–	3
Level of Attainments PEC-CSE414-T															



Web Development

General Information

Course Code: PEC-CSE415-T/ PEC-IT415-T Course Credits: 3 Type: Professional Elective Contact Hours: 3hours/week Mode: Lectures (L) Examination Duration: 3 hours	**C Course Assessment Methods (internal: 30; external: 70)** Two minor examinations (20 marks), Class Performance measured through percentage of lectures attended (4 marks), assignments (6 marks), and the end- semester examination (70 marks). For the end semester examination, nine questions are to be set by the examiner. A candidate is required to attempt 5 questions in all. All questions carry equal marks. Question number 1 will be compulsory and based on the entire syllabus. It will contain seven parts of 2 marks each. Question numbers 2 to 9 will be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt the remaining four questions by selecting one question from each of the four units.

Pre-requisites: knowledge of Computer Basics

About the Course:

Web development is a management of information. Web Development is a core and an essential course for every graduate in Computer Science and Engineering. This course introduces web designing tools like HTML, XML, Java Script and ASP/JSP etc. and various web site will be designed with the help of these tools for solving real world problems. It includes various types of website. Further, It is more useful for dynamic programming as well.

Course Outcomes: By the end of the course students will be able to:

CO 1. **enlist** principles of Information Architecture for Web design. (LOTS: Level 1: Remember)
CO 2. **explain** navigational systems, labeling systems, and taxonomies for websites. (LOTS: Level 2: Understand)
CO 3. **apply** basic web designing tools (HTML, XML, ASP/JSP, JQuery, Java Script). (LOTS: Level 3: Apply) .
CO 4. **evaluate** critically design of webpages based on various technologies. (HOTS: Level 5: Evaluate)
CO 5. **create** a report describing or making recommendations for a website design. (HOTS: Level 6: Create)

Course Content

Unit - I

Information Architecture, Role of Information Architect, Collaboration and Communication, Organizing Information, Organizational Challenges, Organizing Web Sites and Intranets, Creating Cohesive Organization Systems Designing, Navigation Systems, Types of Navigation Systems, Integrated Navigation Elements, Remote Navigation Elements, Designing Elegant Navigation Systems, Searching Systems, Designing the Search Interface, Indexing the Right Stuff, What to Search or not to Search, Grouping Content, Conceptual Design, Architecture Blueprints, Architectural Page Mockups, Design Sketches.

Unit - II

Structured Information, Design and Documentation, XML Web 6.0, JDBC, Metadata, Unstructured Information, Techniques for Unstructured Information, HTML Basic Concepts, Good Web Design, Process of Web Publishing, Phases of Web Site Development, Structure of Html Documents, Html Elements for Designing Pages. Text Level Events, Linking Basics, Linking In Html, Images and Anchors Attributes, Image Maps, Semantic Linking Meta Information, Image Preliminaries, Images, Layout Design, Advanced Layout. Audio Support in Browsers, Video Support, Other Binary Formats. Style Sheets, Positioning With Style Sheets. Basic Interactivity and Html: Forms, Forms Control, Advance HTML and Web Designing.

Unit - III

Alternative Technologies for Designing, The Hypertext Transport Protocol, URLs, HTTP, Browser Requests, Server Responses, Proxies, Content Negotiation, The Common Gateway Interface, The CGI Environment Variables. CGI Output, Forms and CGI, Sending Data to the Server, Form Tags, Decoding Form Input, Architectural Guidelines, Coding Guidelines, Efficiency and Optimization. JSP Basics, Integrating Scripts in JSPs, ASP Objects and Components, JSP: Request and Response Objects, Retrieving the Contents of a HTML form, retrieving a Query String, Cookies, Creating and Reading Cookies.

Unit - IV

XML basics, Relationship between HTML, SGML, and XML, Valid Documents. Ways to use XML, XML for Data Files, Embedding XML into HTML documents, Converting XML to HTML for DISPLAY, Displaying XML using CSS and XSL, Rewriting HTML as XML, Basics of Advance Web Development Tools.

Text and Reference Books:

1. Thomas A Powell, *HTML-The Complete Reference*, Tata McGraw Hill, 2003.
2. Scott Guelich, Shishir Gundavaram, Gunther Birzniek, *CGI Programming with Perl* 2nd edition, O'Reilly, 2000.
3. Doug Tidwell, James Snell, Pavel Kulchenko, *Programming Web Services with SOAP*, O'Reilly, 2009.
4. Young, *XML Step by Step*, 2nd edition, PHI.
5. Paul J. Deitel, Harvey M. Deitel, Abbey Deitel, *Internet & World Wide Web How to Program*, 5th edition, 2008.

CO-PO Articulation Matrix Web Development Course (PEC-CSE415-T)

List of Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1. **Enlist** principles of Information Architecture for Web design. (LOTS: Level 1: Remember)	1	–	–	–	–	–	–	–	–	–	–	–	2	–	–
CO2. **Explain** navigational systems, labeling systems, and taxonomies for websites. (LOTS: Level 2: Understand)	1	–	–	–	–	–	–	–	–	–	–	–	3	–	–
CO3. **Apply** basic web designing tools (HTML, XML, ASP/JSP, JQuery, Java Script). (LOTS: Level 3: Apply)	2	2	2	2	3	3	–	–	–	–	–	–	3	2	–
CO4. **Evaluate** critically design of webpages based on various technologies. (HOTS: Level 5: Evaluate)	3	3	2	3	3	3	–	–	–	–	–	–	3	2	–
CO5. **Create** a report describing or making recommendations for a website design. (HOTS: Level 6: Create)	–	–	–	3	3	–	–	3	3	3	2	–	3	–	–
Level of Attainments PEC-CSE415-T															

Statistical Computing

General Course Information

Course Code: PEC-CSE416-T/ PEC-IT416-T Course Credits: 3 Type: Professional Elective Contact Hours: 3 hours/week Mode: Lectures (L) Examination Duration: 3 hours	**Course Assessment Methods (internal: 30; external: 70)** Two minor examinations (20 marks), Class Performance measured through percentage of lectures attended (4 marks), assignments (6 marks), and the end- semester examination (70 marks). For the end semester examination, nine questions are to be set by the examiner. A candidate is required to attempt 5 questions in all. All questions carry equal marks. Question number 1 will be compulsory and based on the entire syllabus. It will contain seven parts of 2 marks each. Question numbers 2 to 9 will be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt the remaining four questions by selecting one question from each of the four units.

Pre-requisites: Basics of probability

About the Course:

It is important to know essentials of statistics to become a successful data analyst or researcher. This course is tailored to introduce the graduating engineering to the fundamentals of statistics so that they can analyze data and draw inference from it.

Course Outcomes: By the end of the course students will be able to:

CO1. **define** basic tools of data analysis. (LOTS: Level 1: Remember)
CO2. **explain** the concepts given in descriptive and inferential statistics (LOTS: Level 2: Understand)
CO3. **apply** statistical concepts to solve real world statistical computing problems. (LOTS: Level 3: Apply)
CO4. **analyse** the trends in data using descriptive statistics. (HOTS: Level 4: Analyse)
CO5. **interpret and evaluate** statistical models. (HOTS: Level 5: Evaluate)
CO6. **conclude** the findings of statistical analysis. (HOTS: Level 6: Create)

Course Content

Unit I

Review of Descriptive Statistics and Probability Theory: Scale of measurement and data types, Descriptive statistics, Frequency Tables and graphs, Relative frequency tables and graphs, grouping data, histograms and ogive, mean, median, mode, variance and standard deviation of sample data, sample spaces and events, Axioms, Conditional Probability, Independent event, Bayes Theorem, Binomial Theorem.

Unit II

Random Variable and Distributions: Random variables, type of random variables, Mean (Expectation) and variance of a discrete random variables, Discrete uniform distribution, Bernoulli's distribution, Binomial distribution, Geometric distribution, Poisson's distribution, Mean and variance of a continuous random variable, Continuous uniform distribution: normal distribution, exponential distribution, Central Limit Theorem.

Unit III

Hypothesis testing: determining levels of significance, Types of hypothesis testing errors, Hypothesis testing for population mean for large and small samples; Comparing two population means for large and small independent samples; Comparing two population means for paired samples; Comparing two population proportions, Chi-Square, t test and F test, Analysis of variance (ANOVA).

Unit IV

Statistical Learning and Linear Regression: Definition of statistical learning, Estimating a function f, The trade of between prediction accuracy and model comprehensibility, Regression versus Classification problems, Measuring the quality of fit, Bias and Variance trade off, Linear Regression between variables, Estimating the Coefficients, accessing the accuracy of the coefficient estimates, assessing the accuracy of the model, Multiple linear regression, estimating the multiple regression.

Text and Reference Books:

1. Ross Sheldon M., *Introduction to Probability and Statistics for Engineers and Scientists*, 4th edition, Academic Press, 2009.
2. Douglas S. Shafer and Zhang Zhiyi, *Beginning Statistics*, 2012. [Available freely online under Creative Commons by-nc-sa 3.0 license]
3. Brain S. Everitt, *A Handbook of Statistical Analysis Using R*, Second Edition, LLC 2014
4. Roger D. Peng, *R Programming for Data Science*, Lean Publishing, 2015.
5. Michael J. Crawley, *Statistics, An introduction using R*, Second edition, John Wiley, 2015
6. Trevor Hastie, Robert Tibshirani, Jerome Friedman, *The Elements of Statistical Learning: Data Mining, Inference and Prediction*, Springer, 2nd edition, 2009.

CO-PO Articulation Matrix Statistical Computing Course (PEC-CSE-416-T)

List of Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1. **Define** basic tools of data analysis. (LOTS: Level 1: Remember)	1	–	–	–	–	–	–	–	–	–	–	–	–	–	3
CO2. **Explain** the concepts given in descriptive and inferential statistics (LOTS: Level 2: Understand)	1	–	–	–	–	–	–	–	–	–	–	–	–	–	3
CO3. **Apply** statistical concepts to solve real world statistical computing problems. (LOTS: Level 3: Apply)	2	2	2	2	3	–	–	–	–	–	–	–	–	–	3
CO4. **Analyse** the trends in data using descriptive statistics. (HOTS: Level 4: Analyse)	2	3	2	3	3	–	–	–	–	–	–	–	–	–	3
CO5. **Interpret and evaluate** statistical models. (HOTS: Level 5: Evaluate)	2	2	2	3	3	–	–	–	–	–	–	–	–	–	3
CO6. **Conclude** the findings of statistical analysis. (HOTS: Level 6: Create	2	3	2	3	–	–	–	–	–	–	–	–	–	–	3
Level of Attainments PEC-CSE416-T															

Digital Forensics

General Course Information

Course Code: PEC-CSE417-T/ PEC-IT406-T Course Credits: 3 Type: Professional Elective Contact Hours: 3hours/week Mode: Lectures (L) Examination Duration: 3 hours	**Course Assessment Methods (internal: 30; external: 70)** Two minor examinations (20 marks), Class Performance measured through percentage of lectures attended (4 marks), assignments (6 marks), and the end- semester examination (70 marks). For the end semester examination, nine questions are to be set by the examiner. A candidate is required to attempt 5 questions in all. All questions carry equal marks. Question number 1 will be compulsory and based on the entire syllabus. It will contain seven parts of 2 marks each. Question numbers 2 to 9 will be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt the remaining four questions by selecting one question from each of the four units.

Pre-requisites: working knowledge of Windows/Macintosh/Linux, Network security.

About the Course:

The course on Digital Forensics is an inevitable study in this information era. Computer crimes are on a hike by the hackers and cyber criminals. The need to recover the deleted, hidden and corrupted files on Windows/Macintosh/Linux platforms give an opportunity to offer digital forensics automating features. This will give students a chance to study laws of court against computer crimes committed intentionally or inadvertently.

Course outcomes: By the end of the course students will be able to:

CO1. **determine** the hardware and operating system requirements for digital forensics.(LOTS: Level 1: Remember)
CO2. **represent** digital forensics by organization of data and metadata in computer systems.(LOTS: Level 2: Understand)
CO3. **analyze** file recovery and hidden file extraction techniques. (HOTS: Level 4: Analyze)
CO4. **identify** various types of forensics in the arena of information technology. (HOTS: Level 4:Analyze)
CO5. **critic** the computer crimes by studying the security Laws and legal Landscape around the world.(HOTS: Level 5: Evaluate)
CO6. **integrate** security of computer systems with digital forensics and evaluate its performance. (HOTS: Level 6: create)

Course Content

Unit I

Introduction to Digital Forensics: digital crimes, digital investigation, evidence, extraction, preservation etc.; overview of hardware and operating systems: structure of storage media/devices, Windows/Macintosh/Linux-registry, boot process; disk and file system analysis, data acquisition of physical storage devices.

Unit II

Data recovery: identifying hidden data, recovering deleted files; digital evidence controls: uncovering attacks that evade detection by event viewer, task manager and other windows GUI tools; disk imaging, recovering swap files, temporary and cache files; automating analysis and extending capabilities.

Unit III

Network Forensics: collecting and analyzing network-based evidence, reconstructing web browsing, email activity, intrusion detection, tracking offenders, windows registry changes, etc.; Mobile Network forensics: introduction, investigations, collecting evidences, where to seek digital data for further investigations; Email and database forensics; memory acquisition.

Unit IV

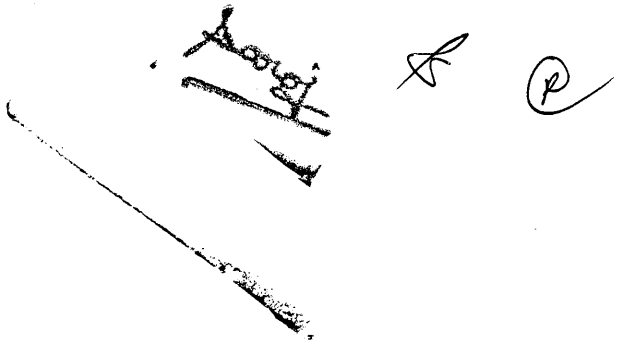
Computer crime and legal issues: intellectual property, privacy issues, criminal justice system for forensic, audit/investigative situations and digital crime scene, investigative procedure/standards for extraction, preservation and deposition of legal evidence in a court of law.

Text and Reference Books:

1. Thomas J Holt , Adam M Bossler, Kathryn C Seigfried-Spellar, *Cybercrime and Digital Forensics: An Introduction*, Routledge, 2015.
2. Cory Altheide and Harlan Carvey, *Digital Forensics with Open Source Tools,* Elsevier publication, April 2011.
3. B. Nelson, A. Phillips, F. Enfinger, C. Steuart, *Guide to Computer Forensics and Investigations* 4th edition, Thomson, 2009.
4. Michael Hale Ligh, Andrew Case, Jamie Levy, AAron Walters, *The Art of Memory Forensics: Detecting Malware and Threats in Windows, Linux, and Mac Memory*, july 2014.

CO-PO Articulation Matrix Digital Forensics Course (PEC-CSE417-T)

List of Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1. **Determine** the hardware and operating system requirements for digital forensics. (LOTS: Level 1: Remember)	1	–	–	–	–	–	–	–	–	–	–	–	–	2	–
CO2. **Represent** digital forensics by organization of data and metadata in computer systems. (LOTS: Level 2: Understand)	1	–	–	–	–	–	–	–	–	–	–	–	–	3	–
CO3. **Analyze** file recovery and hidden file extraction techniques. (HOTS: Level 4: Analyze)	2	2	–	–	3	–	–	–	–	–	–	–	–	3	–
CO4. **Identify** various types of forensics in the arena of information technology. (HOTS: Level 4: Analyze)	2	2	2	2	3	–	–	–	–	–	–	–	–	3	2
CO5. **Critic** the computer crimes by studying the security Laws and legal Landscape around the world. .(HOTS: Level 5: Evaluate)	3	3	3	3	–	3	–	3	–	3	–	–	–	3	–
CO6. Integrate security of computer systems with digital forensics and evaluate its performance. (HOTS: Level 6: create)	3	3	2	3	3	–	–	–	–	–	–	–	–	3	–
Level of Attainments PEC-CSE417-T															



Internet of Things Lab.

General Course Information

Course Code: PEC-CSE409-P/ PEC-IT409-P Course Credits: 1 Type: Professional Elective Lab. Course Contact Hours: 2 hours/week Mode: Lab practice and assignments	**Course Assessment Methods (internal: 30; external: 70)** The internal assessment is based on the percentage of lab sessions attended (4 marks), timely submission of lab experiments/assignments and the quality of solutions provided in the assignments (16 marks), and an internal VIVA-VOCE (10 marks) conducted towards the end of semester. The external examination is of 70 marks. The break-up of marks for external examination is based on quality of lab reports (20 marks), quality of solution(s) for the given problem(s) at the time of examination (written work + execution of program(s)) (30) and VIVA-VOCE examination (20).

Pre-requisites: Basic knowledge of C/C++ language, Basics of Electronics.

About the Course:

This course focuses on significant components of Internet of Things. The objective of this lab course is to make the students familiar with prototype and key components of networking for development of application based on Internet of Things.

Course outcomes: By the end of the lab course students will be able to:

CO1. **solve** the existing problems of traditional sensor networks and wireless communication using the concepts of Internet of Things. (LOTS: Level 3: Apply)
CO2. **analyse** the working of controllers and sensors. (HOTS: Level 4: Analyse)
CO3. **compare** and contrast the existing solutions related to IOT. (HOTS: Level 5: Evaluate)
CO4. **design** solutions for practical assignments by using Internet of Things technologies. (HOTS: Level 6: Create)
CO5. **create** lab reports by presenting the ideas regarding solutions in an effective manner. (HOTS: Level 6: Create)
CO6. **demonstrate** independent enquiry, team spirit and ethical practices while solving problems. (LOTS: Level 3: Apply)

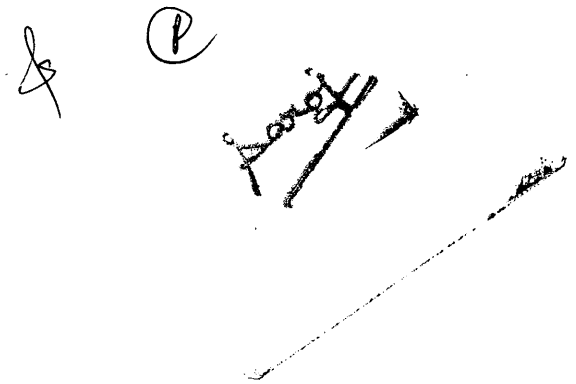
List of experiments/assignments:

1. In order to implement IoT practical assignments one needs the following:
   - Hardware Setup- device capable of storage and network, e.g. Raspberry Pi, Intel Galileo, Intel, Edison, Multiple sensors etc.
   - Software- Wiring Pi (C++ for Raspberry Pi), Wiring x86 (Python for Intel Edison)
   - API to connect hardware to web server
   - Web Interface
2. Two assignments to figure out input and output devices.
3. Two assignments to interface digital and analogue devices with microcontroller unit.
4. Two assignment for calibration of sensors.
5. Two assignments for receiving data from sensors serially.
6. Two assignments to read the values from sensors.

7. Two assignments based on testing of temperature sensor, integrating of temperature sensor with microcontroller, temperature control over internet.

Note:

The actual experiments/assignments will be designed by the course coordinator. One assignment should be designed to be done in groups of two or three students. The assignments must meet the objective of the course and the levels of the given course outcomes. The list of assignments and schedule of submission will be prepared by the course coordinator at the beginning of the semester.



CO-PO Articulation Matrix Internet of Things Lab. Course (PEC-CSE409-P)

List of Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1. **Solve** the existing problems of traditional sensor networks and wireless communication using the concepts of Internet of Things. (LOTS: Level 3: Apply)	2	2	2	2	3	–	–	–	–	–	–	–	–	3	2
CO2. **Analyse** the working of controllers and sensors. (HOTS: Level 4: Analyse)	2	2	2	3	3	–	–	–	–	–	–	–	–	3	2
CO3. **Compare** and contrast the existing solutions related to IOT. (HOTS: Level 5: Evaluate)	3	2	2	3	2	–	–	–	–	–	–	–	–	3	2
CO4. **Combining** Internet of Things technologies for designing solutions for complex problems. (HOTS: Level 6: Create)	3	3	2	3	3	–	–	–	–	–	–	–	–	3	3
CO5. **Create** lab reports by presenting the ideas regarding solutions in an effective manner. (HOTS: Level 6: Create)	–	–	–	–	–	–	–	–	–	3	–	–	–	–	–
CO6. **Demonstrate** independent enquiry, team spirit and ethical practices while solving problems. (LOTS: Level 3: Apply)	–	–	–	–	–	–	–	3	3	–	3	–	–	–	–
Level of Attainments PEC-CSE409-P															

Software Defined Networks Lab.

General Course Information

Course Code: PEC-CSE410-P/ PEC-IT410-P Course Credits: 1 Type: Professional Elective Lab. Course Contact Hours: 2 hours/week Mode: Lab practice and assignments	**Course Assessment Methods (internal: 30; external: 70)** The internal and external assessment is based on the level of participation in lab. sessions and the timely submission of lab experiments/assignments, the quality of solutions designed for the assignments, the performance in VIVA-VOCE, the quality of lab. file and ethical practices followed. The internal examination is conducted by the course coordinator. The external examination is conducted by external examiner appointed by the Controller of Examination in association with the internal examiner appointed by the Chairperson of the Department.

Pre-requisites: knowledge of Computer Networks and Java.

About the Course:

This course on Software Defined Networks is a development lab. which involves configuration of open switches for different platforms. It incorporates setting up of hosts to be connected to a Network through SDN Controllers installed on servers. The objective of the lab course is to equip the students to solve the issues related to Openflow protocol through OpenFlow Standard, Mininet and OpenDaylight Controllers.

Course Outcomes: By the end of the course students will be able to:

CO1. **implement** SDN controllers using API/mininet. (LOTS: Level 3: Apply)
CO2. **analyse** results of SDN statistics for a given scenario. (HOTS: Level 4: Analyse)
CO3. **assess** performance of protocols for a given Network. . (HOTS: Level 5: Evaluate)
CO4. **hypothesize** solutions for SDN controller issues by using Network statistics. (HOTS: Level 6: Create)
CO5. **create** lab records for the assignment solutions. (HOTS: Level 6: Create)
CO6. **demonstrate** use of ethical practices, self-learning and team spirit. (LOTS: Level 3: Apply)

List of experiments/assignments

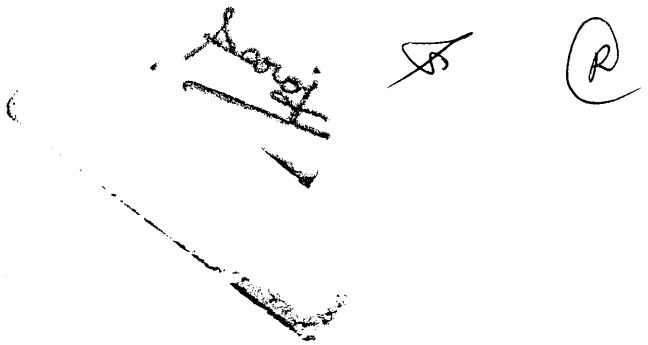
1. Introduction to the OpenFlow Standard.
2. SDN Controller concepts and interfaces using Open Day light Controller (Java based implementation, REST interface, OSGI module interface)
3. Implementation of centralized static and dynamic routing protocols.
4. Control plane distribution for increased availability and scalability.
5. Open Day light in an open Networking Platform that enables SDN and constructs a solid foundation for Network functions Virtualization for all Network sizes. Perform virtualization in the Data Centre and in the Network.
6. Open Day light Controller- Brokers and RPC calls, the Datastore, plugin Development workflow, Development environment setup.
7. Testing and performance evaluation using software switches (open vSwitch), hardware switches and Network emulation (Mininet). (in group of 2-3 students)

Note:

The actual experiments/assignments will be designed by the course coordinator. One assignment should be designed to be done in groups of two or three students. The assignments must meet the objective of the course and the levels of the given course outcomes. The list of assignments and schedule of submission will be prepared by the course coordinator at the beginning of the semester.

CO-PO Articulation Matrix Software Defined Networks Lab. Course (PEC-CSE410-P)

List of Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1. **Implement** SDN controllers using API/mininet. (LOTS: Level 3: Apply)	2	2	2	2	3	–	–	–	–	–	–	–	–	3	–
CO2. **Analyse** results of SDN statistics for a given scenario. (HOTS: Level 4: Analyse)	2	3	2	3	3	–	–	–	–	–	–	–	–	3	–
CO3. **Assess** performance of protocols for a given Network. . (HOTS: Level 5: Evaluate)	3	3	2	3	3	–	–	–	–	–	–	–	–	3	–
CO4. **Hypothesize** solutions for SDN controller issues by using Network statistics. (HOTS: Level 6: Create)	3	3	2	3	3	–	–	–	–	–	–	–	–	3	–
CO5. **Create** lab records for the assignment solutions. (HOTS: Level 6: Create)	–	–	–	–	–	–	–	–	–	3	–	–	–	–	–
CO6. **Demonstrate** use of ethical practices, self-learning and team spirit. (LOTS: Level 3: Apply)	–	–	–	–	–	–	–	3	3	–	–	3	–	–	–
Level of Attainments PEC-CSE410-P															



Network Administration and Management Lab.

General Course Information

Course Code: PEC-CSE411-P/ PCC-IT305-P Course Credits: 1 Type: Professional Elective Lab. Course Contact Hours: 2 hours/week Mode: Lab practice and assignments	**Course Assessment Methods (internal: 30; external: 70)** The internal and external assessment is based on the level of participation in lab. sessions and the timely submission of lab experiments/assignments, the quality of solutions designed for the assignments, the performance in VIVA-VOCE, the quality of lab. file and ethical practices followed. The internal examination is conducted by the course coordinator. The external examination is conducted by external examiner appointed by the Controller of Examination in association with the internal examiner appointed by the Chairperson of the Department.

Pre-requisites: knowledge of Computer Networks, System Administration, Unix/Linux Command line.

About the Course:

This lab. course on Network Administration and Management involves configuration of servers for different platforms. It incorporates setting up of ones' machine to be connected to a Network and checking its status frequently for any intrusion. The objective of the lab. course is to equip the students to solve the practical Administration, Management and Monitoring related problems.

Course Outcomes: By the end of the course students will be able to:

CO1. **configure** a server to work as a DNS/DHCP/FTP/Web/Mail/Print server (LOTS: Level 3: Apply)
CO2. **detect** the trends in attacks through in depth attack analysis. (HOTS: Level 4: Analyse)
CO3. **formulate** solutions for Monitoring assignments by using principles of Network statistics. (HOTS: Level 6: Create)
CO4. **plan** solutions for overall security of Computer/Network systems. (HOTS: Level 6: Create)
CO5. **create** file records of solutions of assignments. (HOTS: Level 6: Create)
CO6. **demonstrate** use of ethical practices, self-learning and team spirit. (LOTS: Level 3: Apply)

List of experiments/assignments:

1. Management (creation, modification and deletion of left users) of the users & their domain.
2. Setting up the local security policy for the system, software.
3. Maintaining your system in Linux Networking and Setup Linux for firewall and IP filtering.
4. Configure the kernel for IP Accounting and IP Masquerade.
5. Install sendmail distribution and create sendmail configuration files.
6. Start and stop services from user window and command prompt.
7. Use of event viewer and performance monitor.
8. Management of the IIS and FTP server.
9. Setting up of router in Window 2000 server and Linux server.
10. Use of utilities (a) Ping (b) Tracert (c) netstat(d) net (e) IP configuration (f) Path ping

11. Monitor the Network using performance monitoring tools such as RMON, tcpdump etc.
12. Setting up of a DNS server.
13. Setting up and use "Terminal Client Services".

Note:

The actual experiments/assignments will be designed by the course coordinator. One assignment should be designed to be done in groups of two or three students. The assignments must meet the objective of the course and the levels of the given course outcomes. The list of assignments and schedule of submission will be prepared by the course coordinator at the beginning of the semester.

CO-PO Articulation Matrix Network Administration and Management Lab. Course (PEC-CSE411-P)

List of Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1. **Configure** a server to work as a DNS/DHCP/FTP/Web/Mail/Print server (LOTS: Level 3: Apply)	1	–	2	–	3	–	–	–	–	–	–	–	–	3	–
CO2. **Detect** the trends in attacks through in depth attack analysis. (HOTS: Level 4: Analyse)	2	2	2	2	3	–	–	–	–	–	–	–	–	3	–
CO3. **Formulate** solutions for Monitoring assignments by using principles of Network statistics. (HOTS: Level 6: Create)	3	3	3	3	3	–	–	–	–	–	–	–	–	3	–
CO4. **Plan** solutions for overall security of Computer/Network systems. (HOTS: Level 6: Create)	3	3	3	3	3	2	–	–	–	–	–	–	–	3	–
CO5. **Create** file records of solutions of assignments. (HOTS: Level 6: Create)	–	–	–	–	–	–	–	–	–	3	–	–	–	–	–
CO6. **Demonstrate** use of ethical practices, self-learning and team spirit. (LOTS: Level 3: Apply)	–	–	–	–	–	–	–	3	3	–	–	3	–	–	–
Level of Attainments PEC-CSE411-P															

Software Testing and Quality Assurance Lab.

General Course Information

Course Code: PEC-CSE412-P/ PEC-IT412-P Course Credits: 1 Type: Professional Elective Lab. Course Contact Hours: 2 hours/week Mode: Lab practice and assignments	**Course Assessment Methods (internal: 30; external: 70)** The internal and external assessment is based on the level of participation in lab. sessions and the timely submission of lab experiments/assignments, the quality of solutions designed for the assignments, the performance in VIVA-VOCE, the quality of lab. file and ethical practices followed. The internal examination is conducted by the course coordinator. The external examination is conducted by external examiner appointed by the Controller of Examination in association with the internal examiner appointed by the Chairperson of the Department.

Pre-requisites: Knowledge of Software Engineering along with Programming in C/C++/Java or /MATLAB.

About the Course:

In this lab. Course, students learn to design, generate, minimize, and prioritize test cases of a software application using programming language or with the help of software testing tools. The lab experiments involve designing testing datasets by taking case studies and applying software testing techniques on these datasets. The course has a special focus on understanding and implementation of test results of software testing techniques to improve software quality.

Course Outcomes: By the end of the course students will be able to:

CO1. **implement** software testing using testing tools. (LOTS: Level 3: Apply)
CO2. **apply** software testing techniques for the classification of test cases. (LOTS: Level 3: Apply)
CO3. **interpret** the results of various software testing techniques. (HOTS: Level 4: Analyse)
CO4. **plan** test case activities. (HOTS: Level 6: Create)
CO5. **prepare** lab reports for software quality testing assignments. (HOTS: Level 6: Create)
CO6. **demonstrate** use of ethical practices, self-learning and team spirit. (LOTS: Level 3: Apply)

List of experiments/assignments

1. Write a program to count the number of digits in a number. Its input is any number from interval [0, 9999]. Design the boundary value analysis test cases and robustness test cases.
2. Write a program to calculate cyclomatic complexity.
3. Consider a program to perform binary search and generate the test cases using equivalence class testing and decision table based testing.
4. Write a program to determine whether a number is even or odd. Draw the program graph and DD path graph. Find the independent paths.
5. Consider the program for classification of a triangle. Consider all variables and generate possible program slices. Design at least one test case from every slice.
6. Consider the problem statement of a University Student Registration System. Prepare the software requirement checklist with the details of faults in the given SRS.

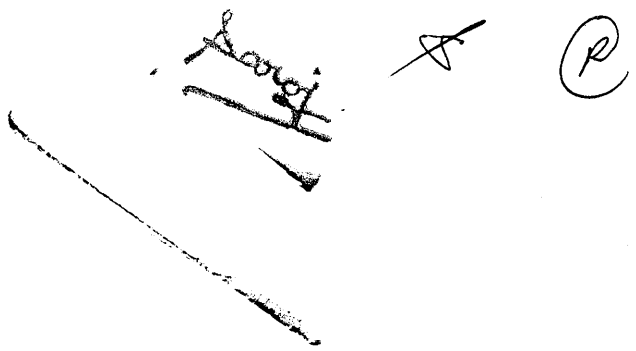
7. Write a program to generate, minimize and prioritize test cases using any programming language/Matlab Tool/Software Testing tool.
8. Write the outline of test plan document as per IEEE Std 829-1998.
9. One assignment to be done in groups.

Note:

The actual experiments/assignments will be designed by the course coordinator. One assignment should be designed to be done in groups of two or three students. The assignments must meet the objective of the course and the levels of the given course outcomes. The list of assignments and schedule of submission will be prepared by the course coordinator at the beginning of the semester.

CO-PO Articulation Matrix Software Testing and Quality Assurance Lab. Course (PEC-CSE412-P)

List of Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1. **Implement** software testing using testing tools. (LOTS: Level 3: Apply)	2	2	2	2	3	–	–	–	–	–	–	–	3	–	–
CO2. **Apply** software testing techniques for the classification of test cases. (LOTS: Level 3: Apply)	2	2	2	2	3	–	–	–	–	–	–	–	3	–	–
CO3. **Interpret** the results of various software testing techniques. (HOTS: Level 4: Analyse)	3	2	3	3		–	–	–	–	–	–	–	3	–	–
CO4. **Plan** test case activities. (HOTS: Level 6: Create)	3	3	3	3	3	–	–	–	–	–	3	–	3	–	–
CO5. **Prepare** lab reports for software quality testing assignments. (HOTS: Level 6: Create)	–	–	–	–	–	–	–	–	–	3	–	–	–	–	–
CO6. **Demonstrate** use of ethical practices, self-learning and team spirit. (LOTS: Level 3: Apply)	–	–	–	–	–	–	–	3	3	–	–	3	–	–	–
Level of Attainments PEC-CSE412-P														–	–



Machine Learning Lab.

General Course Information

Course Code: PEC-CSE-413-P/ PEC-IT413-P Course Credits: 1 Type: Professional Elective Lab. Course Contact Hours: 2 hours/week Mode: Lab practice and assignments	**Course Assessment Methods (internal: 30; external: 70)** The internal and external assessment is based on the level of participation in lab. sessions and the timely submission of lab experiments/assignments, the quality of solutions designed for the assignments, the performance in VIVA-VOCE, the quality of lab. file and ethical practices followed. The internal examination is conducted by the course coordinator. The external examination is conducted by external examiner appointed by the Controller of Examination in association with the internal examiner appointed by the Chairperson of the Department.

Pre-requisites: Programming in Java, Python, R and Octave/MATLAB.

About the Course:

In this lab. Course, students learn to solve optimization, supervised and unsupervised learning problems using machine learning tools. Students will use machine learning tools available in WEKA, R, Python and Octave etc. The lab experiments involve downloading datasets and applying machine learning techniques on these datasets. The course has a special focus on interpreting and visualizing results of machine learning algorithms.

Course Outcomes: By the end of the course students will be able to:

CO1. **implement** machine learning algorithms using modern machine learning tools. (LOTS: Level 3: Apply)
CO2. **analyse** the trends in datasets using descriptive statistics. (HOTS: Level 4: Analyse)
CO3. **apply** descriptive and predictive modelling. (LOTS: Level 3: Apply)
CO4. **compare and contrast** machine learning algorithms for a given problem. (describe datasets using descriptive statistics. (HOTS: Level 5: Evaluate)
CO5. **create** lab records of assignment by incorporating problem definitions, design of solutions, results and interpretations. (HOTS: Level 6: Create)
CO6. **demonstrate** use of ethical practices, self-learning and team spirit. (LOTS: Level 3: Apply)

List of experiments/assignments

1. Install WEKA/R/Python/Octave and learn to use these software packages.
2. Two assignments related to classification algorithms and interpreting the results of these algorithms.
3. Two assignments related to clustering algorithms and interpreting the results of these algorithms.
4. Three assignment on designing neural networks for solving learning problems.
5. Two assignment on ranking or selecting relevant features.
6. Two assignments on linear regression and logistic regression.
7. One assignment to be done in groups.

Note:

The actual experiments/assignments will be designed by the course coordinator. One assignment should be designed to be done in groups of two or three students. The assignments must meet the objective of the course and the levels of the given course outcomes. The list of assignments and schedule of submission will be prepared by the course coordinator at the beginning of the semester.

CO-PO Articulation Matrix Machine Learning Lab. Course (PEC-CSE413-P)

List of Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1. **Implement** machine learning algorithms using modern machine learning tools. (LOTS: Level 3: Apply)	2	2	2	3	3	–	–	–	–	–	–	–	–	–	3
CO2. **Analyse** the trends in datasets using descriptive statistics. (HOTS: Level 4: Analyse)	3	3	2	3	3	–	–	–	–	–	–	–	–	–	3
CO3. **Apply** descriptive and predictive modelling. (LOTS: Level 3: Apply)	3	3	2	3	3	–	–	–	–	–	–	–	–	–	3
CO4. **Compare and contrast** machine learning algorithms for a given problem. (describe datasets using descriptive statistics. (HOTS: Level 5: Evaluate)	3	3	2	3	3	–	–	–	–	–	–	–	–	–	3
CO5. **Create** lab records of assignment by incorporating problem definitions, design of solutions, results and interpretations. (HOTS: Level 6: Create)	–	–	–	–	–	–	–	–	–	3	–	–	–	–	–
CO6. **Demonstrate** use of ethical practices, self-learning and team spirit. (LOTS: Level 3: Apply)	–	–	–	–	–	–	–	3	3	–	–	3	–	–	–
Level of Attainments PEC-CSE413-P															

Big Data Analytics Lab.

General Course Information

Course Code: PEC-CSE414-P/ PEC-IT414-P Course Credits: 1 Type: Professional Core Lab. Course Contact Hours: 2 hours/week Mode: Lab practice and assignments	**Course Assessment Methods (internal: 30; external: 70)** The internal and external assessment is based on the level of participation in lab. sessions and the timely submission of lab experiments/assignments, the quality of solutions designed for the assignments, the performance in VIVA-VOCE, the quality of lab. file and ethical practices followed. The internal examination is conducted by the course coordinator. The external examination is conducted by external examiner appointed by the Controller of Examination in association with the internal examiner appointed by the Chairperson of the Department.

Pre-requisites: Some basic knowledge and experience of Java (JARS, Array, Classes, Objects, etc.)

About the Course:

This lab course provides an overview of key technology used in manipulating, storing, and analyzing big data. This incorporates big data analytics and use of Hadoop.

Course Outcomes: By the end of the course students will be able to:

CO1. **implement** solutions for big data problem. (LOTS: Level 3: Apply)
CO2. **apply** Hadoop ecosystem components. (LOTS: Level 3: Apply)
CO3. **analyse** the results of big data algorithms. (HOTS: Level 4: Analyse)
CO4. **build** and maintain reliable, scalable, distributed systems. (HOTS: Level 6: Create)
CO5. **create** lab record of the lab assignments that contains problem definitions, their solutions in big data perspective and the interpretation of the results. (HOTS: Level 6: Create)
CO6. **demonstrate** ethical practices, self-learning and team spirit. (LOTS: Level 3: Apply)

List of experiments/assignments

1. Installing and configuring Hadoop cluster.
2. Manipulating files in HDFS using Hadoop fs commands.
3. Hadoop File Systems: IBM GPFS, MapR-FS, Lustre, Amazon S3 etc.
4. Writing an Inverted Index MapReduce Application.
5. Distributed Cache MapReduce Design Patterns Sorting Joins.
6. Writing a streaming MapReduce job in Hadoop.
7. Big Data and R: Clustering, Simple Linear Regression, Decision Trees, Naïve Bayesian Classification
8. Big Data Interactions: Big Data and Cloud: Big Data and Web Services /SOA:Big Data and Internet of Things (IoT)
9. Big Data Case Study: Healthcare Data: Web Click stream Data: Social Media Data [RSS, Tweets]

Note:

The actual experiments/assignments will be designed by the course coordinator. One assignment should be designed to be done in groups of two or three students. The assignments must meet the objective of the course and the levels of the given course outcomes. The list of assignments and schedule of submission will be prepared by the course coordinator at the beginning of the semester.

CO-PO Articulation Matrix Big Data Analytics Lab. Course (PEC-CSE414-P)

List of Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1. **Implement** solutions for big data problem. (LOTS: Level 3: Apply)	2	2	2	2	3	–	–	–	–	–	–	–	–	–	3
CO2. **Apply** Hadoop ecosystem components. (LOTS: Level 3: Apply)	2	2	2	2	3	–	–	–	–	–	–	–	–	–	3
CO3. **Analyse** the results of big data algorithms. (HOTS: Level 4: Analyse)	3	2	2	3	3	–	–	–	–	–	–	–	–	–	3
CO4. **Build** and maintain reliable, scalable, distributed systems. (HOTS: Level 6: Create)	2	3	3	3	3	–	–	–	–	–	–	–	–	–	3
CO5. **Create** lab record of the lab assignments that contains problem definitions, their solutions in big data perspective and the interpretation of the results. (HOTS: Level 6: Create)	–	–	–	–	–	–	–	–	–	3	–	–	–	–	–
CO6. **Demonstrate** ethical practices, self-learning and team spirit. (LOTS: Level 3: Apply)	–	–	–	–	–	–	–	3	3	–	–	3	–	–	–
Level of Attainments PEC-CSE414-P															

Web Development Lab.

General Course Information

Course Code: PEC-CSE415-P/ PEC-IT415-P Course Credits: 1 Type: Professional Elective Lab Course Contact Hours: 2 hours/week Mode: Lab practice and assignments	**Course Assessment Methods (internal: 30; external: 70)** The internal and external assessment is based on the level of participation in lab. sessions and the timely submission of lab experiments/assignments, the quality of solutions designed for the assignments, the performance in VIVA-VOCE, the quality of lab. file and ethical practices followed. The internal examination is conducted by the course coordinator. The external examination is conducted by external examiner appointed by the Controller of Examination in association with the internal examiner appointed by the Chairperson of the Department.

Pre-requisites: Basic programming skills and knowledge of surfing internet.

About the Course:

This lab. course on web development involves learning web-based programming languages. It incorporates the development of web pages by structuring information provided for the website design. The objective of the lab course is to equip the students to design web pages using modern web development tools.

Course Outcomes: By the end of the course students will be able to:

CO1. **implement** object models for website design using modern tools like HTML, XMLand JAVA scripting etc. (LOTS: Level 3: Apply)
CO2. **analyse** the design of websites. (HOTS: Level 4: Analyse)
CO3. **test** the design of websites. (HOTS: Level 5: Evaluate)
CO4. **design** websites that consider socio-cultural values. (HOTS: Level 6: Create)
CO5. **create** a written report for website designed. (HOTS: Level 6: Create)
CO6. **use** ethical practices and socio-cultural values while designing websites. (LOTS: Level 3: Apply)

List of experiments/assignments

1. Create a simple webpage using HTML.
2. Designing of registration form with table and use of hyperlink.
3. Design a page with frames to include Images and Videos.
4. Add a cascading style sheet for designing the web page.
5. Use user defined function to get array of values and sort them in ascending order on web page
6. Design a dynamic web page with validation of form field using JavaScript.
7. Design a catalogue in ASP.
8. Event Handling Validation of registration form.
9. Open a Window from the current window on Mouse Over event.
10. Create a simple application to demonstrate Servlets Request and Response object.
11. Demonstrate Array Objects and Date Object's predefined methods

12. Display calendar for the month and year selected from combo box
13. Create a welcome Cookie (Hit for a page) and display different image and text content each time when the user hit the page
14. Demonstrate Request and Response object using HTML Form.
15. Database Connection to display all the values in the table.

Note:

The actual experiments/assignments will be designed by the course coordinator. One assignment should be designed to be done in groups of two or three students. The assignments must meet the objective of the course and the levels of the given course outcomes. The list of assignments and schedule of submission will be prepared by the course coordinator at the beginning of the semester.

CO-PO Articulation Matrix Web Development Lab. Course (PEC-CSE415-P)

List of Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1. **Implement** object models for website design using modern tools like HTML, XMLand JAVA scripting etc. (LOTS: Level 3: Apply)	2	2	2	–	3	–	–	–	–	–	–	–	3	–	–
CO2. **Analyse** the design of websites. (HOTS: Level 4: Analyse)	3	3	2	2	3	–	–	–	–	–	–	–	3	–	–
CO3. **Test** the design of websites. (HOTS: Level 5: Evaluate)	2	3	2	2	3	–	–	–	–	–	–	–	3	–	–
CO4. **Design** websites that consider socio-cultural values. (HOTS: Level 6: Create)	3	3	3	3	3	3	–	–	–	–	–	–	3	–	–
CO5. **Create** a written report for website designed. (HOTS: Level 6: Create)	–	–	–	–	–	–	–	–	–	3	–	–	–	–	–
CO6. **Use** ethical practices and socio-cultural values while designing websites. (LOTS: Level 3: Apply)	–	–	–	–	–	–	–	3	3	–	–	3	–	–	–
Level of Attainments PEC-CSE415-P															

Statistical Computing Lab.

General Course Information

Course Code: PEC-CSE416-P/ PEC-IT416-P Course Credits: 1 Type: Professional Elective Lab. Course Contact Hours: 2 hours/week Mode: Lab practice and assignments	**Course Assessment Methods (internal: 30; external: 70)** The internal and external assessment is based on the level of participation in lab. sessions and the timely submission of lab experiments/assignments, the quality of solutions designed for the assignments, the performance in VIVA-VOCE, the quality of lab. file and ethical practices followed. The internal examination is conducted by the course coordinator. The external examination is conducted by external examiner appointed by the Controller of Examination in association with the internal examiner appointed by the Chairperson of the Department.

Pre-requisites: Basic Statistics and Programming in Python, R

About the Course:

In this lab. Course, students learn to solve statistical computing problems using R or Python. The lab experiments involve applying statistical tools for analyzing and inferring information from real world datasets. The course has a special focus on interpreting, evaluating and concluding from the results of statistical analysis.

Course Outcomes: By the end of the course students will be able to:

CO1. **implement** statistical tools for drawing inference from data. (LOTS: Level 3: Apply)
CO2. **explore** the trends in datasets using descriptive statistics. (HOTS: Level 4: Analyse)
CO3. **apply** probability, hypothesis testing and regression for solving research questions. (LOTS: Level 3: Apply)
CO4. **Judge** different problem situations for applying appropriate statistical tests (HOTS: Level 5: Evaluate)
CO5. **create** lab records of assignment by incorporating problem definitions, design of solutions, results and interpretations. (HOTS: Level 6: Create)
CO6. **demonstrate** use of ethical practices, self-learning and team spirit. (LOTS: Level 3: Apply)

List of experiments/assignments

1. Install R and R studio.
2. Two assignments related to descriptive statistics.
3. Two assignments related to visualizing trends in data.
4. Three assignments related to permutations, combinations and probability.
5. Four assignments on Hypothesis Testing.
6. Two assignments on linear regression.
7. Two assignments on logistic regression.
8. One assignment to be done in groups.

Note:

The actual experiments/assignments will be designed by the course coordinator. One assignment should be designed to be done in groups of two or three students. The assignments must meet the objective of the course and the levels of the given course outcomes. The list of assignments and schedule of submission will be prepared by the course coordinator at the beginning of the semester.

CO-PO Articulation Matrix Statistical Computing Lab. Course (PEC-CSE416-P)

List of Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1. **Implement** statistical tools for drawing inference from data. (LOTS: Level 3: Apply)	2	2	2	3	3	–	–	–	–	–	–	–	–	–	3
CO2. **Explore** the trends in datasets using descriptive statistics. (HOTS: Level 4: Analyse)	2	2	2	2	3	–	–	–	–	–	–	–	–	–	3
CO3. **Apply** probability, hypothesis testing and regression for solving research questions. (LOTS: Level 3: Apply)	2	3	2	3	3	–	–	–	–	–	–	–	–	–	3
CO4. **Judge** different problem situations for applying appropriate statistical tests (HOTS: Level 5: Evaluate)	3	3	3	3	3	–	–	–	–	–	–	–	–	–	3
CO5. **Create** lab records of assignment by incorporating problem definitions, design of solutions, results and interpretations. (HOTS: Level 6: Create)	–	–	–	–	–	–	–	–	–	3	–	–	–	–	–
CO6. **Demonstrate** use of ethical practices, self-learning and team spirit. (LOTS: Level 3: Apply	–	–	–	–	–	–	–	3	3	–	–	3	–	–	–
Level of Attainments PEC-CSE416-P															

Digital Forensics Lab.

General Course Information

Course Code: PEC-CSE417-P/ PEC-IT406-P Course Credits: 1 Type: Professional Elective Lab. Course Contact Hours: 2 hours/week Mode: Lab practice and assignments	**Course Assessment Methods (internal: 30; external: 70)** The internal assessment is based on the percentage of lab sessions attended (4 marks), timely submission of lab experiments/assignments and the quality of solutions provided in the assignments (16 marks), and an internal VIVA-VOCE (10 marks) conducted towards the end of semester. The external examination is of 70 marks. The break-up of marks for external examination is based on quality of lab reports (20 marks), quality of solution(s) for the given problem(s) at the time of examination (written work + execution of program(s)) (30) and VIVA-VOCE examination (20).

Pre-requisites: The students are expected to have a knowledge of components of computer system, operating systems like Windows, Macintosh, Linux.

About the Course:

This course on Digital Forensics is a developmental laboratory work. It incorporates file system recovery related to various operating systems. The objective of the lab course is to equip the students to solve the practical digital forensics issues.

Course outcomes: By the end of the lab course student will be able to:

CO1. **employ** the digital forensics tools for file system analysis. (LOTS: level 3: Apply)
CO2. **test** ethical practices while solving the problems at hand. (HOTS: level 4: Analyze)
CO3. **select** open source tools for imaging various types of media by wiping a target drive. (HOTS: level 5: evaluate)
CO4. **develop** solutions for disk imaging and like problems in different hardware conditions and for various operating systems. (HOTS: level 6: create)
CO5. **design** Lab record for the assignments including aim, hardware and software requirements and solutions to given problems. (HOTS: Level 6: Create)
CO6. **demonstrate** independent enquiry, use of ethical practices and self-learning to solve unseen problems. (LOTS: level 2: understand)

List of experiments/assignments:

1. Two assignments on forensically examining Window registry for evidences located in it.
2. Two assignments on wiping a target drive and ensure that it is wiped, imaging various types of media such as hard drives, USB flash drives, optical drives, ZIP disks.
3. Two assignments on system restore points and how they are valuable in a forensic investigation.
4. Two assignments on open source tool autopsy for timeline analysis, hash filtering and file system analysis.
5. Two-three assignments on open source tool caine for mobile forensics, Network forensics, data recovery.
6. Two-three assignments on Helix3 for incident response and computer forensics.

Note:

The actual experiments/assignments will be designed by the course coordinator. One assignment should be designed to be done in groups of two or three students. The assignments must meet the objective of the course and the levels of the given course outcomes. The list of assignments and schedule of submission will be prepared by the course coordinator at the beginning of the semester.

CO-PO Articulation Matrix Digital Forensics lab. Course (PEC-CSE417-P)

List of Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1. **Employ** the digital forensics tools for file system analysis. (LOTS: level 3: Apply)	2	2	–	3	3	–	–	–	–	–	–	–	–	3	–
CO2. **Deploy** forensics techniques/algorithms in solving problems related to the arena of Engineering/Information Technology. (LOTS: Level 3: Apply)	2	3	2	2	3	–	–	–	–	–	–	–	–	3	–
CO3. **Select** open source tools for imaging various types of media by wiping a target drive. (HOTS: level 5: evaluate)	2	2	3	3	3	–	–	–	–	–	–	–	–	3	–
CO4. **Develop** solutions for system restore points and like problems in different hardware conditions and for various operating systems. (HOTS: level 6: create)	3	3	3	3	3	–	–	–	–	–	–	–	–	3	–
CO5. **Design** Lab record for the assignments including aim, hardware and software requirements and solutions to given problems. (HOTS: Level 6: Create)	–	–	–	–	–	–	–	–	–	3	–	–	–	–	–
CO6. **Demonstrate** independent enquiry, use of ethical practices and self-learning to solve unseen problems. (LOTS: level 2: understand)	–	–	–	–	–	–	–	3	3	–	–	3	–	–	–
Level of Attainments PEC-CSE417-P															

Major Project II

General Project Information

Course Code: PROJ-CSE403 Course Credits: 6 Mode: Self learning under the guidance of a faculty member.	**Course Assessment Methods (Internal evaluation: 30 marks; External Evaluation marks: 70)** Evaluation is done by the internal examiner (project guide) and external examiner appointed by Controller of Examination. The criteria for evaluation are given below. 1. Review of literature related to problem domain: 15 2. Significance and originality of the solution presented: 15 3. Application of software engineering principles and project management: 15 4. Significance and Scope of results: 20 5. Organisation and presentation of major project report: 20 6. Level of Ethics and societal issues covered: 15

About the major project II:

Students continue working on their project work and they are required to complete their project work by the end of 8th semester. Students carry out implementation of their respective projects based on the problem identified, methodology and tools suggested in the synopsis prepared during seventh semester. They prepare the final project reports according to the format provided. At the end of eighth semester, each student is required to present his/her project work in front of internal project guide and external examiner appointed by Controller of Examination.

Course Outcomes: After doing major Project students will be able to:

CO1. **review** information critically for solving complex engineering problems. (HOTS: Level 4: Analyse)
CO2. **plan** the project according to principles of project management. (HOTS: Level 6: Create)
CO3. **devise** original solutions to complex engineering problems using modern engineering tools. (HOTS: Level 6: Create)
CO4. **justify** the outcomes of the project work. (HOTS: Level 5: Evaluate)
CO5. **organise** and communicate (written and oral) ideas effectively. (HOTS: Level 6: Create)
CO6. **develop** solutions that meet ethical, societal and legal considerations. (HOTS: Level 6: Create)

CO-PO Articulation Matrix Course (PROJ-CSE403)

List of Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1. **Review** information critically for solving complex engineering problems. (HOTS: Level 4: Analyse)	2	2	2	3										
CO2. **Plan** the project according to principles of project management. (HOTS: Level 6: Create)	1	1	1								3	3		
CO3. **Devise** original solutions to complex engineering problems using modern engineering tools. (HOTS: Level 6: Create)	3	2	3	3	3	2								
CO4. **Justify** the outcomes of the project work. (HOTS: Level 5: Evaluate)	3	3	3	3		2								
CO5. **Organise** and communicate (written and oral) ideas effectively. (HOTS: Level 6: Create)										3	3			
CO6. **Develop** solutions that meet ethical, societal and legal considerations. (HOTS: Level 6: Create)								3	3			3		
Level of Attainments PROJ-CSE403