Curriculum and Syllabus

Bachelor of Technology

(Computer Science and Engineering)

(As per National Education Policy-2020)

Department of Computer Science and Engineering

School of Engineering and Technology,

H. N. B. Garhwal University, Srinagar Garhwal, Uttarakhand- 246174

ADMISSION PROCEDURE AND ELIGIBILITY

The minimum qualification for admission to the B. Tech. First year shall be higher secondary school certificate examination (10+2) scheme with Physics, Chemistry and Mathematics or any other subject notified by the UGC/AICTE conducted by Central Board of Secondary Education (CBSE) or any other equivalent examination from recognized board or university. The candidate shall be eligible for admission to B. Tech. Degree Course based on CUET score. In general, the admission to B. Tech. degree course shall be governed by the rules of, Ministry of Education (MOE), Government of India (GOI) and Hemvati Nandan Bahuguna Garhwal University (A Central University), Srinagar Garhwal, Uttarakhand.

B. Tech (Computer Science and Engineering)	Physics, Mathematics and Chemistry/Computer Science/Electronics/Information Technology/ Biology/Informatics Practices/ Biotechnology	**Direct Entry-** For B. Tech. I semester: Candidate shall have passed 10+2 examination from any recognized Board/University with Physics and Mathematics as a compulsory subject along with Chemistry/Computer Science/Electronics/Information Technology/ Biology/Informatics Practices/ Biotechnology obtained at least 45% marks (40% in case of candidate belonging to reserved category) in the above subjects taken together. Candidates seeking admission to B.Tech. Course (All branches) in HNB Garhwal University have to appear for CUET (UG). They are required to opt for Physics, Mathematics and Chemistry/Computer Science/Electronics /Information Technology/ Biology/Informatics Practices/ Biotechnology for the admission in HNBGU. Candidates who will not appear in Physics, Mathematics and Chemistry/Computer Science/Electronics/Information Technology/ Biology/Informatics Practices/Biotechnology will not be considered for admission in B.Tech. [Criterion for preparation of Merit list: CUET (UG) merit list: **After the CUET (UG) merit list is exhausted; the vacant seats may be filled up by the candidates from the merit list of the university level counselling on the basis of JEE Main's Score merit.**

Admission to **B. Tech. 2nd year**, will be made in two modes:

(i) Students who have passed a three-year Diploma in Engineering recognised by the AICTE/State University or B.Sc. in the relevant/respective discipline will be eligible for admission as per admission procedure adopted by the university time to time.

(ii) Students who have obtained an equivalent Vocational/Technical Program/Under Graduate Certificate with NHEQF Level 5/ UCF-Level 4.5 will be eligible for admission to 3rd Semester in B. Tech. in the respective discipline of engineering degree program against the vacant seats. However, the candidate shall be eligible for admission on the basis of the entrance examination conducted or any other procedure adopted by Hemvati Nandan Bahuguna Garhwal University for admission to B. Tech. Degree course time to time.

Admission to B. Tech. 3rd year shall be held if the candidate passed B. Tech. 2nd year along with NHEQF Level 6 /UCF-Level 5 equivalent to Undergraduate Diploma in the respective discipline. The admission to B. Tech.

(Fifth Semester) degree program will be held against the vacant seats. However, the candidate shall be eligible for admission on the basis of the entrance examination conducted or any other procedure adopted by Hemvati Nandan Bahuguna Garhwal University for admission to B. Tech. Degree course time to time.

Admission to B. Tech. Fourth year shall be held if the candidate passed B. Tech. 3rd year along with Advanced Diploma/ Bachelor degree in Vocational Education in relevant discipline with NHEQF Level 7/UCF-Level 5.5. The admission to B. Tech. (Seventh Semester) degree program will be held against the vacant seats. However, the candidate shall be eligible for admission on the basis of the entrance examination conducted or any other procedure adopted by Hemvati Nandan Bahuguna Garhwal University for admission to B. Tech. Degree course time to time. With regard to the qualification earned from foreign country, an equivalence certificate from the university /Association of Indian Universities is mandatory for admission to B. Tech. Programs. In case of any dispute about the equivalence in qualification earned from foreign countries, the decision of university in this regards shall be final and binding on all concerned.

The reservation in admission, cancellation of admission and fee refund will be as per AICTE / MOE /GOI/ UGC/ HNBGU norms and notification issued in this regard from time to time.

❖ Eligibility for Under Graduate Certificate

(a) The Under Graduate Certificate course will be offered by all departments of SOET HNB Garhwal University.

(b) All the candidates who have successfully passed one-year (two semester) of B. Tech. Programme will be eligible for Under Graduate Certificate. The candidate who wants to exit with Under Graduate Certificate he/she have to do skill-based training of two months. This skill based training will be of 4 credit based on the job/course specific internship/apprentice-ship equivalent to **NHEQF level 5 / UCF- level 4.5**. The evaluation of the 04 credit will be based on examination/ presentation before the external expert.

(c) A student shall report the concerned Head on or before the date notified by the Department /School, if he/she is interested to exit with Under Graduate Certificate.

❖ Eligibility for Under Graduate Diploma

(a) The Under Graduate Diploma will be offered by all departments of SOET HNB Garhwal University.

(b) All the candidates who have successfully passed two-year (Four semester) of B. Tech. Programme will be eligible for Under Graduate Diploma. The candidate who wants to exit with Under Graduate Diploma shall have to do training of skill-based courses of two months. This skill based training will be of 4 credit based on the job/course specific internship/apprentice-ship equivalent to **NHEQF level 6 / UCF- level 5.** The evaluation of the 04 credit will be based on examination/ presentation before the external expert.

(c) A student shall report the concerned Head on or before the date notified by the Department/School, if he/she is interested to exit with Under Graduate Diploma.

❖ Eligibility for Advance Diploma/ Bachelor of Vocational Education

(a) The Advance Diploma/ Bachelor of Vocational Education will be offered by all departments of SOET HNB Garhwal University.

(b) All the candidates who have successfully passed Three-year (Six semester) of B. Tech. Programme will be eligible

Advance Diploma/ Bachelor of Vocational Education. The candidate who wants to exit with Advance Diploma/ Bachelor of Vocational Education shall have to do training of skill-based courses of two months. This skill based training will be of 4 credit based on the job/course specific internship/apprentice-ship equivalent to **NHEQF level 7/ UCF level 5.5**. The evaluation of the 04 credit will be based on examination/ presentation before the external expert.

(c) A student shall report the concerned Head on or before the date notified by the Department/School/University if he/she is interested to exit with Advance Diploma/ Bachelor of Vocational Education.

- **Eligibility for B. Tech. Degree**

After the Successful completion of the four year B.tech course, a student will be eligible to get B. Tech. Degree in the concerned branch.

Note: The definition of NHEQF levels / UCF levels can be obtained from web-link

https://www.ugc.gov.in/pdfnews/2142241_NHEQF-Draft.pdf

Curriculum

General Instructions

Mandatory Induction Program

3 weeks duration
❖ Physical activity ❖ Creative Arts ❖ Universal Human Values ❖ Literary ❖ Proficiency Modules ❖ Lectures by Eminent People ❖ Visits to local Areas ❖ Familiarization to Dept./Branch & Innovations

Induction program for students is to be offered right at the start of the first year. Appendix –I sheet has attached for details.

1. **Code for Courses:**

   Code for a course consists of two alphabets followed by three digits and an optional alphabet.

   - First three alphabets represent the school name (SET: School of Engineering and Technology).
   - Next two alphabets in the code represent the subject area of the course, e.g., SH: Applied Science and Humanities, EC: Electronics and Communication Engineering, IE: Instrumentation Engineering, EE: Electrical Engineering, ME: Mechanical Engineering, CS: Computer Science and Engineering, IT: Information Technology.
   - Next two alphabets in the code represent the name of program, e.g., BT: B. Tech., MT: M. Tech.
   - Then there will be subject code with 4 letters out of which first will tell the nature of subject (C: Core/ E: Elective/ S: Skill/ M: Mandatory/ L: Life Skills and Personality Development) and next three letters will tell the number according to the semester (for example 801 will tell its 8^{th} semester subject). First digit represents the semester. Next two digits represent the sequence number of course in the list of courses of a semester.
   - The code of the common syllabus for UG courses of the university will be elaborated as VAC: Value Added Course, AMDSC: Additional Multi-Disciplinary Skill Course.

2. **Elective Courses:**

   Elective courses are provided in V, VI, VII and VIII semesters to provide student with flexibility to choose courses of their interest from a list of offered electives. These Electives are the courses offered by the same department or other departments for the students.

3. **MOOC Courses:** "MOOCs" means Massive Open Online Courses (MOOCs) are such online courses which are developed and made available on the SWAYAM platform of Government of India. MOOCs guidelines on online learning issued by the MHRD vide orders dated 11th March 2016 and subsequent addendums issued by the MHRD.

   Any student can be permitted to opt for only up to 20% of the total courses being offered in a particular program in a semester through the online learning courses provided through SWAYAM platform. Any student can opt, with the permission of the department, the course of the SWAYAM platform, which is available/ offered in the same term (even or odd).

4. For the slow learners, a remedial class of 1 hr., on each working day will be conducted.
5. Entry/Exit options after 1st year, 2nd year, and 3rd year, respectively may be allowed as per the admission procedure and eligibility of the School of Engineering and Technology (SOET).

All the courses and course titles are subject to change at any stage as per directions of Authorities of the University.

Appendix-I

Mandatory Induction Program
B.Tech (Computer Science and Engineering)

3 weeks duration
❖ Physical activity ❖ Creative Arts ❖ Universal Human Values ❖ Literary ❖ Proficiency Modules ❖ Lectures by Eminent People ❖ Visits to local Areas ❖ Familiarization to Dept./Branch & Innovations

***Induction program for students to be offered right at the start of the first year.**

Induction Program:

When new students enter an institution, they come with diverse thoughts, backgrounds and preparations. It is important to help them adjust to the new environment and inculcate in them the ethos of the institution with a sense of larger purpose. Precious little is done by most of the institutions, except for an orientation program lasting a couple of days.

We propose a 3-week long induction program for the UG students entering the institution, right at the start. Its purpose is to make the students feel comfortable in their new environment, open them up, set a healthy daily routine, create bonding in the batch as well as between faculty and students, develop awareness, sensitivity and understanding of the self, people around them, society at large, and nature.

The time during the Induction Program is also used to rectify some critical lacunas, for example, English background, for those students who have deficiency in it. The following are the activities under the induction program in which the student would be fully engaged throughout the day for the entire duration of the program.

❖ Physical Activity:

This would involve a daily routine of physical activity with games and sports. It would start with all students coming to the field at 6 am for light physical exercise or yoga. There would also be games in the evening or at other suitable times according to the local climate. These would help develop team work. Each student should pick one game and learn it for three weeks. There could also be gardening or other suitably designed activity where labour yields fruits from nature.

❖ Creative Arts:

Every student would choose one skill related to the arts whether visual arts or performing arts. Examples are painting, sculpture, pottery, music, dance etc. The student would pursue it every day for the duration of the program. These would allow for creative expression. It would develop a sense of aesthetics and also enhance creativity which would, hopefully, flow into engineering design later.

❖ Universal Human Values:

It gets the student to explore oneself and allows one to experience the joy of learning, stand up to peer pressure, take decisions with courage, be aware of relationships with colleagues and supporting staff in the hostel and department, be sensitive to others, etc.

Need for character building has been underlined earlier. A module in Universal Human Values provides the base. Methodology of teaching this content is extremely important. It must not be through do's and dont's, but get students to explore and think by engaging them in a dialogue. It is best taught through group discussions and real life activities rather than lecturing. The role of group discussions, however, with clarity of thought of the teachers cannot be over emphasized. It is essential for giving exposure, guiding thoughts, and realizing values. Discussions would be conducted in small groups of about 20 students with a faculty mentor each. It is to open thinking towards the self. Universal Human Values discussions could even continue for rest of the semester as a normal course, and not stop with the induction program. Besides drawing the attention of the student to larger issues of life, it would build relationships between teachers and students which last for their entire 4-year stay and possibly beyond.

❖ Literary:

Literary activity would encompass reading, writing and possibly, debating, enacting a play etc.

❖ Proficiency Modules:

This period can be used to overcome some critical lacunas that students might have, for example, English, computer familiarity etc. These should run like crash courses, so that when normal courses start after the induction program, the student has overcome the lacunas substantially. We hope that problems arising due to lack of English skills, wherein students start lagging behind or failing in several subjects, for no fault of theirs, would, hopefully, become a thing of the past.

❖ Lectures by Eminent People:

This period can be utilized for lectures by eminent people, say, once a week. It would give the students exposure to people who are socially active or in public life.

❖ Visits to Local Area:

A couple of visits to the landmarks of the city, or a hospital or orphanage could be organized. This would familiarize them with the area as well as expose them to the under privileged.

❖ Familiarization to Dept. /Branch & Innovations:

The students should be told about different method of study compared to coaching that is needed at IITs. They should be told about what getting into a branch or department means what role it plays in society, through its technology. They should also be shown the laboratories, workshops & other facilities.

PROGRAM OUTCOMES (POS) - ENGINEERING & TECHNOLOGY

Engineering Graduates will be able to:

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 analyze 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 modeling 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- B.Tech. (CSE)

At the end of the program

PSO1: Graduates will have adequate knowledge of computer engineering domain to become employable in Industry.

PSO2: Graduate will have strong fundamentals and problem solving skills to analyze, design and develop economically feasible solutions for technical and social problems.

PSO3: Graduate will be aware of recent research trends, higher education and entrepreneurial opportunities, and will work ethically towards society.

PSO4: Graduate will be aware about the latest technology in software and hard ware.

PSO5: Graduate will be exposed to industrial training giving hands on experience.

Semester- wise List of Subjects (As per NEP 2020)

Semester I

S. No.	Category	Course Code	Course Code and Title	L	T	P	Contact Hrs./Week	Credits
1	Basic Science/Multidisciplinary	SET/SH/BT/C101	Mathematics I	3	1	-	4	4
2		SET/SH/BT/C103	Chemistry	3	1	-	4	4
3	Core Basic Engineering Subjects	SET/ME/BT/C104	Engineering Mechanics	3	1	-	4	4
4		SET/ME/BT/C102	Basic Mechanical Engineering	3	1	-	4	4
5		SET/CS/BT/C105	C Programming	3	1		4	4
6	Core/Basic Engineering Subjects Labs	SET/SH/BT/C108	Chemistry Lab	-		1	3	1
7		SET/CS/BT/C109	C Programming Lab			1	3	1
8	Extracurricular Courses/CC	VAC-1	Understanding and Connecting Student with Environment*	2	-	-	2	2
9	Skill Course	SET/CS/SC/C110	Internet Technology Lab-I (Skill Enhancement Course)	-	-	1	4	2
Total				17	5	3	32	26

***L= Lectures, T=Tutorials , P= Practicals**

***** Common syllabus for all UG Courses of the University.

Semester II

S. No.	Category	Course Code	Course Code and Title	L	T	P	Contact Hrs./Week	Credits
1	Basic Science/Multidisciplinary	SET/SH/BT/C201	Mathematics II	3	1	-	4	4
2		SET/SH/BT/C202	Physics	3	1	-	4	4
3	Core Basic Engineering Subjects	SET/EE/BT/C203	Basic Electrical Engineering	3	1	-	4	4
4		SET/EC/BT/C204	Basic Electronics	3	1	-	4	4
5		SET/IT/BT/C205	Fundamental of Information Technology	3	1		4	4
6	Core/Basic Subjects Based Labs	SET/SH/BT/C207	Physics Lab	-		1	3	1
7		SET/ME/BT/C208	Engineering Graphics and Workshop Practice			1	3	1
8	Life Skills and personality development	VAC-2	A Life Skills and personality Development*	2	-	-	2	2
9	Skill Course	SET/CS/SC/C210 or SET/CS/SC/C211	Internet Technology Lab-II or Basics of Python Lab (Skill Enhancement Course)	-	-	1	4	2
Total				17	5	3	32	26

***L= Lectures, T=Tutorials , P= Practicals**

***** Common syllabus for all UG Courses of the University.

Entry/Exit Option with Under Graduate Certificate in Computer Science and Engineering, may be allowed after completion of the 1st year, and as per requirements defined in the admission procedure and eligibility of the School..

Semester III

S. No.	Category	Course Code	Course Code and Title	L	T	P	Contact Hrs./Week	Credits
1	Basic Science/Multidisciplinary	SET/AH/BT/C301	Mathematics III	3	1	-	4	4
2	Core Subjects	SET/CS/BT/C302	Computer Based Numerical & Statistical Techniques	3	1	-	4	4
3		SET/CS/BT/C304	Data Structures Using C	3	1	-	4	4
4		SET/CS/BT/C305	Discrete Structures	3	1	-	4	4
5	Interdisciplinary Subject	SET/EC/BT/C303	Digital Electronics	3	1		4	4
6	Core Subjects Based Labs	SET/CS/BT/C306	Computer Based Numerical & Statistical Techniques lab	-		1	3	1
7		SET/CS/BT/C307	Digital Electronics Lab			1	3	1
8	Extracurricular Courses/CC	**AMDSC-2**	***Additional Multidisciplinary Skill course (AMSC): Any one of the following** **1. Nursery training course** **2. Basic Yoga practices** **3.Physical education/sports management** **4. Folk and culture** **5. Indian traditional music**	2	-	-	2	2
9	Skill Course	SET/CS/SC/C308	Data Structures Using C Lab	-	-	1	4	2
Total				17	5	3	32	26

***L= Lectures, T=Tutorials , P= Practicals**

***** Common Syllabus for all U.G. courses of the University.

Semester IV

S. No.	Category	Course Code	Course Title	L	T	P	Contact Hrs./Week	Credits
1	Core Subjects	SET/CS/BT/C401	Object Oriented Programming using C++	3	1	-	4	4
2		SET/CS/BT/C402	Operating System	3	1	-	4	4
3		SET/CS/BT/C403	Computer Organization and Architecture	3	1	-	4	4
4		SET/CS/BT/C405	Theory of Computation	3	1	-	4	4
5	Interdisciplinary Subject	SET/CS/BT/C404	Data Communication and Computer Network	3	1		4	4
6	Core Subjects Based Labs	SET/CS/BT/C406	Object Oriented Programming using C++ Lab	-		1	3	1
7		SET/CS/BT/C407	Operating System Lab			1	3	1
8	Indian Knowledge System-I (IKS-2)	VAC-3	**Indian Knowledge System***	2	-	-	2	2
9	Skill Course	SET/CS/SC/C408	Mini Project (Based on C/C++)			1	4	2
		Total		17	5	3	32	26

***L= Lectures, T=Tutorials , P= Practicals**

* Common Syllabus for all U.G. courses of the University.

Entry/Exit Option with Under Graduate Diploma in Computer Science and Engineering, may be allowed after completion of the 2nd year, and as per requirements defined in the admission procedure and eligibility of the School.

Semester V

S. No.	Category	Course Code	Course Title	L	T	P	Contact Hrs./Week	Credits
1	Core Subjects	SET/CS/BT/C501	Database Management System	3	1	-	4	4
2		SET/CS/BT/C502	Design and Analysis of Algorithms	3	1	-	4	4
3		SET/CS/BT/C503	Software Engineering	3	1	-	4	4
4			Program Elective-1	3	1	-	4	4
5	Open Elective/Inter-disciplinary Subject		Open Elective-1	3	1		4	4
6	Core Subjects Based Labs	SET/CS/BT/C504	Database Management System Lab	-		1	3	1
7		SET/CS/BT/C505	Design and Analysis of Algorithms Lab			1	3	1
8	Extracurricular Courses/Compulsory course	SET/CS/BT/M506	*Culture, traditions and moral values/ Yoga Practices	-	-	1	4	2
9	Skill Course	SET/CS/SC/C507	Python Lab			1	4	2
Total				15	5	4	34	26

***L= Lectures, T=Tutorials , P= Practicals**

* University will prepare a course with focus on Indian/ Regional culture studies. In case no syllabus is prepared by the university then Yogabhyas course will be offered.

Elective and Open Elective Courses

S. No.	Category	Course Code	Course Title	L	T	P	Contact Hrs./Week	Credits
1	Program Elective-1	SET/CS/BT/E501	Distributed Computing	3	1	-	4	4
2		SET/CS/BT/E502	Graph Theory	3	1	-	4	4
3		SET/CS/BT/E503	Principles of Programming Languages	3	1	-	4	4
1	Open Elective-1	SET/CS/BT/OE501	Java Programming	2	1	1	4	4
2		SET/CS/BT/OE502	Project Management	3	1		4	4
3		SET/CS/BT/OE503	Optimization Techniques	3	1		4	4

Semester VI

S. No.	Category	Course Code	Course Title	L	T	P	Contact Hrs./Week	Credits
1	Core Subjects	SET/CS/BT/C601	Compiler Design	3	1	-	4	4
2		SET/CS/BT/C602	Computer Graphics	3	1	-	4	4
3		SET/CS/BT/C603	Cryptography and Network Security	3	1	-	4	4
4			Program Elective-2	3	1	-	4	4
5	Open Elective/Inter-disciplinary Subject		Open Elective-2	3	1		4	4
6	Core Subjects Based Labs	SET/CS/BT/C604	Compiler Designing Lab	-		1	3	1
7		SET/CS/BT/C605	Computer Graphics Lab			1	3	1
8	Communication skills/CC	SET/CS/BT/M606	Communication skill Course* /Technical Seminar	-	-	1	4	2
9	Skill Course	SET/CS/SC/C607	Mini Project			1	4	2
Total				15	5	4	34	26

***L= Lectures, T=Tutorials , P= Practicals**

* University will prepare communication courses in Modern/Indian languages from which student will select one language course. The course will be more on applied side with giving students a chance to develop their soft skills. In case no syllabus is prepared by the university then Technical Seminar course will be offered

Program Elective and Open Elective Courses

S. No.	Category	Course Code	Course Title	L	T	P	Contact Hrs./Week	Credits
1	**Program Elective-2**	SET/CS/BT/E601	Data Mining and Data Warehousing	3	1	-	4	4
2		SET/CS/BT/E602	E-Commerce	3	1	-	4	4
3		SET/CS/BT/E603	Data Science	3	1	-	4	4
1	**Open Elective-2**	SET/CS/BT/OE601	Robotic Engineering	3	1	-	4	4
2		SET/CS/BT/OE602	Web Technology	3	1		4	4
3		SET/CS/BT/OE603	Digital Image Processing	3	1		4	4

Note: Entry/Exit Option with Advanced Diploma/ Bachelor degree in Vocational Education (Computer Science and Engineering), may be allowed after completion of **SSD course work** in any one semester within one to six semesters and successfully completing three years, and as per requirements defined in the admission procedure and eligibility of the School.

Semester VII

S. No.	**Category**	**Course Code**	**Course Title**	L	T	P	Contact Hrs./Week	Credits
1	Core Subjects	SET/CS/BT/C701	Artificial Intelligence	3	1	-	4	4
2			Program Elective-3	3	1	-	4	4
3			Program Elective-4	3	1	-	4	4
4	Core Subjects Based Labs/Industrial Oriented Training	SET/CS/BT/C702	Artificial Intelligence Lab	-	-	1	3	1
5		SET/CS/BT/S703	Industrial Training Seminar	-	-	1	3	1
6	Life Skills and personality development		Essential Management Practices*	2	-	-	2	2
7	Skill Course	SET/CS/SC/C704	Project Stage-1	-	-	1	8	4
Total				11	3	3	28	20

***L= Lectures, T=Tutorials , P= Practicals**

***** University will prepare a course with focus on essential management practices.

S. No.	**Category**	**Course Code**	**Course Title**	L	T	P	Contact Hrs./Week	Credits
1	**Program Elective-3**	SET/CS/BT/E701	Wireless and Mobile Computing	3	1	-	4	4
2		SET/CS/BT/E702	Security Architecture & Operating System Security	3	1	-	4	4
3		SET/CS/BT/E703	Neural Network	3	1	-	4	4
1	**Program Elective-4**	SET/CS/BT/E704	Real Time System	3	1	-	4	4
2		SET/CS/BT/E705	Cloud Computing	3	1	-	4	4
3		SET/CS/BT/E706	Computer Vision	3	1	-	4	4

Semester VIII

S. No.	Category	Course Code	Course Title	L	T	P	Contact Hrs./Week	Credits
1	Core Subjects	SET/CS/BT/C801	UNIX Shell Programming	3	1		4	4
2			Program Elective-5	3	1	-	4	4
3			Program Elective-6	3	1	-	4	4
4	Life Skills and personality development	SET/CS/BT/L802	Disaster Management	-	-	1	4	2
5	Skill Course	SET/CS/SC/C803	Project and Dissertation			1	12	6
		Total		9	3	2	28	20

***L= Lectures, T=Tutorials , P= Practicals**

S. No.	Category	Course Code	Course Title	L	T	P	Contact Hrs./Week	Credits
1	Program Elective-5	SET/CS/BT/E801	Natural Language Processing	3	1	-	4	4
2		SET/CS/BT/E802	Internet of Things (IoT)	3	1	-	4	4
3		SET/CS/BT/E803	Machine Learning	3	1	-	4	4
4		SET/CS/BT/E804	Big Data Analytics	3	1	-	4	4
1	Program Elective-6	SET/CS/BT/E805	Cyber Security and Ethical Hacking	3	1	-	4	4
2		SET/CS/BT/E806	Mobile Application Development	3	1		4	4
3		SET/CS/BT/E807	Blockchain Technology	3	1		4	4
4		SET/CS/BT/E808	Deep Learning	3	1		4	4

After successful completion of the eight semester program the student will be eligible for the award of B. Tech. degree in Computer Science and Engineering.

Detailed Syllabi

Semester I

S. No.	Category	Course Code	Course Code and Title	L	T	P	Contact Hrs./Week	Credits
1	Basic Science/Multidisciplinary	SET/SH/BT/C101	Mathematics I	3	1	-	4	4
2		SET/SH/BT/C103	Chemistry	3	1	-	4	4
3	Core Basic Engineering Subjects	SET/ME/BT/C104	Engineering Mechanics	3	1	-	4	4
4		SET/ME/BT/C102	Basic Mechanical Engineering	3	1	-	4	4
5		SET/CS/BT/C105	C Programming	3	1		4	4
6	Core/Basic Engineering Subjects Labs	SET/SH/BT/C108	Chemistry Lab	-		1	3	1
7		SET/CS/BT/C109	C Programming Lab			1	3	1
8	Extracurricular Courses/CC	VAC-1	Understanding and Connecting Student with Environment*	2	-	-	2	2
9	Skill Course	SET/CS/SC/C110	Internet Technology Lab-I (Skill Enhancement Course)	-	-	1	4	2
Total				17	5	3	32	26

* Common syllabus for all UC Courses of the University.

SET/SH/BT/C101	MATHEMATICS- I	
Course Objective	1. To impart a deep understanding of the fundamental concepts and techniques of differential calculus, vector calculus, and matrices. 2. To develop the ability to apply these mathematical principles to solve real-world problems in engineering, physics, and other sciences. 3. To enhance analytical and critical thinking skills through the study of calculus and linear algebra. 4. To prepare students for advanced studies and research in mathematics and related fields. 5. To foster a comprehensive understanding of the theoretical and practical aspects of calculus and matrix theory.	
Course Outcome	1. Students will be able to understand and apply the concepts of limit, continuity, and differentiability for single and multivariable functions. 2. Students will gain proficiency in using vector calculus to solve problems involving gradients, divergences, curls, and applying Gauss's, Green's, and Stokes' theorems. 3. Students will be able to perform matrix operations, understand vector spaces, and apply theorems related to matrices such as the Cayley-Hamilton theorem. 4. Students will develop the skills to solve linear systems, determine the rank of matrices, and find eigenvalues and eigenvectors. 5. Students will be able to apply differential and vector calculus and matrix theory to optimize functions and solve practical problems in various disciplines.	
Module Name	**Content**	**No. of Hrs.**
Differential Calculus	Limit, continuity and differentiability of single and two variables, mean value theorems, indeterminate forms; partial derivatives, total derivative, Euler's formula, Taylor series (in one and two variables), maxima and minima, Extrema of function of several variables, Lagrange"s method.	**15**
Vector Calculus	Interpretation of vectors and scalars, directional derivatives, line, surface and volume integrals, gradient, divergence and curl of a vector and their physical interpretation, Gauss's divergence, Green's and Stoke's theorem.	**12**
Matrices	Vector space, basis, matrices, determinants, Elementary row and column transformation, linear dependence and independence, rank of matrix, consistency of system of linear equation and solution of linear system of equations. Characteristic equation, Cayley-Hamilton theorem, Eigen values and Eigen vectors, diagonalization, complex matrices.	**15**
	Total No. of Hrs.	**42**
Textbooks	1. R. K. Jain and S. R. K. Iyengar "Advanced Engineering Mathematics", Narosa Publications, 2. B. S. Grewal, "Higher Engineering Mathematics", Khanna Publishers, 3. H K Das, "Advanced Engineering Mathematics", S Chand, 4. Erwin Kreyszig, "Advanced Engineering Mathematics".	

SET/SH/BT/C103	CHEMISTRY	
Course Objective	1. Apply the electrochemical principles in batteries, understand the fundamentals of corrosion. 2. Analysis of water for its various parameters and its significance in industrial and domestic Applications. 3. Analyse microscopic chemistry in terms of atomic, molecular orbitals and Intermolecular forces 4. Analysis of major chemical reactions that are used in the synthesis of molecules. 5. V. Understand the chemistry of various fuels and their combustion.	
Course Outcome	1. Describe and understand the operation of electrochemical systems for the production of electric energy, i.e. batteries. 2. Explain the mode by which potable water is produced through the processes of screening, micro Straining, aeration, coagulation and flocculation, sedimentation, flotation, filtration and disinfection. 3. Recognize that molecular orbital theory is a method used by chemists to determine the energy of the electron in a molecule as well as its geometry. 4. Demonstrate an ability to design, implement, and evaluate the results of experimentation using standard scientific methodologies such as hypothesis formulation and testing. 5. Understand and analyse the combustion mechanisms of various fuels	
Module Name	**Content**	**No. of Hrs.**
Advanced Theory of Chemical Bonding	Valence bond and molecular orbital theory. Structure of NH3, H2O, SO3, PCl5, XeO2 molecules. Types of linkages, Hybridization, Hydrogen bonding, Metallic bonding.	4
Equilibrium on Reactivity	Bronsted and Lewis Acids, pH, pka, pkb scale, buffer solution.	4
Polymers	Structures of the following polymers, viz, Natural and synthetic rubbers, Polyamide and Polyester fibres, polymethylmethacrylate, poly acrylonitrile and polystyrene. A brief account of conducting polymers (polypyrrole & polytiphene) & their applications.	3
Complex Compounds	Introduction, Valence bond and crystal field theory.	4
Chemical Kinetics & Catalysis	Order of reactions, Parallel and reversible reactions. Catalysis-homogeneous and heterogeneous catalysis. Characteristics of catalytic reactions, catalytic promoters and poisons, auto catalysis and negative catalysis. Activation energy of catalysis, intermediate compound formation theory and adsorption theory.	3
Atmospheric Chemistry& Air Pollution	Environment and ecology, environmental segments, structure and composition of atmosphere, radiation balance of earth and Green House Effect, formation and depletion of Ozone layer, chemical and photochemical reactions of various species in atmosphere, air pollution-sources, reactions and sinks for pollutants, acid rains and smog formation. Pollution control methods.	5
Corrosion & Lubricants	Introduction, causes of corrosion, theories of corrosion- direct chemical attack, electrochemical theory of corrosion, factors influencing corrosion, corrosion inhibitors, passivity, types of corrosions, protection from corrosion and protective coatings. Theory, classification and mechanism of lubrication.	5
Water and Waste Water Chemistry	Introduction, hardness of water, characteristics imparted by impurities, analysis of contaminants, treatment of water by Zeolite, L-S process, boiler feed water, waste water treatment.	6
Fuels & Combustion	Classification of fuels, non-conventional energy, biogas, biomass and solar energy, calorific value – gross and net, characteristics of good fuel, determination of calorific value, solid fuels, analysis of coal, liquid fuels.	5
Stereochemistry of organic-compounds	Mechanism of chemical reaction, Beckman, Hoffman, Reimer Tiemann, Cunnizzaro, Diels- Alder and Skraup synthesis.	3

	Total No. of Hours 42
Textbooks	1. Jain, Jain, "Engineering Chemistry" 2. Sharma, Kumar, "Engineering Chemistry"
References	1. R. T. Morrison and R N Boyd, "Organic Chemistry", 6th Edition, Prentice Hall, New Delhi, 2. J. D. Lee, "Concise Inorganic Chemistry", Chapman & Hall 3. W. L. Jolly, "Modern Inorganic Chemistry", McGraw-Hill 4. P.W. Atkins, "Physical Chemistry", 6th Edition, Oxford University Press 5. Barrow, "Physical Chemistry" 6. Manahan, "Environmental Chemistry"
	7. D. L. Pavia, GM. Lampman, GS. Kriz and J.R Vyvyan, I, "Spectroscopy", Cengage Learning India Pvt. Ltd, New Delhi, 2007 8. R.M. Silverstein, F.X. Webster and D.J. Kiemle, "Spectrometric Identification of Organic Compounds", 7th edition, John-Wiley and Sons, New York, 2005 9. William Kemp, "Organic Spectroscopy", 3rd edition, Palgrave, New York, 2005 10. C.N. Banwell and E. M. McCash, "Fundamentals of Molecular Spectroscopy", McGraw- Hill, International, UK, 1995 11. F. Carey, "Organic Chemistry", 5th Edition, McGraw Hill Publishers, Boston, 2003

SET/ME/BT/C104	ENGINEERING MECHANICS	
Course Objective	1. To understand distributed force systems, centroid/ centre of gravity and method of finding centroids of composite figures and bodies. 2. To understand the moment of inertia and method of finding moment of inertia of areas and bodies. 3. To understand types of frames and analyse for the forces in the members of the truss by method of joints and method of sections. 4. To understand dynamics of a particle. 5. To interpret the simple given dynamic problems and solve them for positions, velocities and accelerations, etc., 6. To understand the kinetics of the rigid bodies and solve simple problems using work-energy method. • To understand virtual work method and solve simple problems.	
Course Outcome	1. Identify the significance of centroid/ centre of gravity and find centroids of composite figures and bodies. 2. Understand the moment of inertia and method of finding moment of inertia of areas and bodies. 3. Identify the type of frame and analyse for the forces in the members of the truss (frame) by method of joints and method of sections. 4. Understand dynamics of a particle. 5. Interpret the simple given dynamic problems and solve them for positions, velocities and accelerations, etc., 6. Understand the kinetics of the rigid bodies and solve simple problems using work-energy method. • Understand virtual work method and solve simple problems.	
Module Name	**Content**	**No. of Hrs.**
Force System	Introduction: Force system, dimensions and units in mechanics, laws of mechanics, vector algebra, addition and subtraction of forces, cross and dot products of vectors, moment of a force about a point and axis, couple and couple moment, transfer of a force to a parallel position, resultant of a force system using vector method, Problems involving vector application Equilibrium: Static and dynamic equilibrium, static in determinacy, general equations of equilibrium, Varingnon's theorem, Lami's theorem, equilibrium of bodies under a force system, Problems.	8
Trusses And Frames	Truss and Frames: Truss, classification of truss, assumptions in truss analysis, perfect truss, analysis of perfect plane truss using method of joints and method of sections, Problems.	8
Centre Of Gravity And Moment Of Inertia	Centroid, Centre of mass and Centre of gravity, Determination of centroid, centre of mass and centre of gravity by integration method of regular and composite figures and solid objects, Problems, Moment of Inertia: Area moment of inertia, mass moment of inertia, parallel axis and perpendicular axis theorems, radius of gyration, polar moment of inertia, product of inertia, principle axis, problem based on composite figures and solid objects.	10
Friction and Virtual Work	Friction-characteristics of dry friction, problems involving friction of ladder, wedges and connected bodies. Definition of virtual work, principle of virtual work for a system of connected bodies	7
Kinematics And Dynamics	Kinematics: Concept of rigid body, velocity and acceleration, relative velocity, translation and rotation of rigid bodies, equations of motion for translation and rotation, problems. Particle Dynamics: Energy methods and momentum methods, Newton's laws, work energy equation for a system of particles, linear and angular momentum equations, projectile motion, problem.	12
	Total No. of Hours	**45**
Textbooks	1. R S Khurmi, "Engineering Mechanics". 2. P K Nag "Engineering Thermodynamics".	

References	1. Van Wylen G.J. & Sonnlog R.E.: Fundamentals of classical thermodynamics, John Wiley & Sons, Inc. NY. 2. Wark Kenneth: Thermodynamics (2nd edition), Mc Graw Hill book Co. NY. 3. Holman, J.P.: Thermodynamics, MC Graw Hill book Co. NY. 4. Yadav R.: Thermodynamics and Heat Engines, Vol I & II (Sl Edition) Central Publishing House Allahabad. 5. Yadav R.: Steam & Gas Turbines. 6. Kshitish Chandra Pal: Heat Power, Orient Longman Limited, 17, Chittranjan Avenue, Calcutta. 7. S. Rao, B.B. Parulekar, 'Energy Technology', Khanna Pub., New Delhi. 8. G. H. Ryder: "Strength of Materials". 9. F. L. Singer: "Strength of Materials". 10. Timoshenko: "Strength of Materials".

SET/ME/BT/C102	BASIC MECHANICAL ENGINEERING	
Course Objective	1. To use mechanical principles to solve real-world engineering issues. 2. To identify appropriate structural system for studying a given problem and isolate it from its environment. 3. Develop a simple mathematical model for an engineering problem and perform a static analysis. 4. To carry out kinematics and Kinetics analysis for practices and system of particles.	
Course Outcome	1. Students will be able to apply and demonstrate the concept of mechanics to practical engineering problems. 2. Students will be able to determine the properties of planes and solids. 3. Students will be able to apply the basic concept of dynamics to practical problems.	
Module Name	**Content**	**No. of Hrs.**
Fundamental concept of thermodynamics	Definition of thermodynamics, System, Surrounding and Universe, Phase, Concept of continuum, Macroscopic & microscopic point of view. Thermodynamic equilibrium, Property, State, Path, Process, Cyclic and non-cyclic processes, Reversible and irreversible processes, Quasi static process, Energy and its forms, Enthalpy, Zeroth law, first law, second law and third law of thermodynamics, Steady flow energy equation, Limitations of first law of thermodynamics, Essence of second law, Thermal reservoir, Heat engines. COP of heat pump and refrigerator, Carnot cycle, Carnot theorem, Clausius inequality, Concept of entropy.	8
Properties of gases and steam	Boyle's law, Charles's law, Gay-Lussac's law, Avogadro's law, Combined gas law, Gas constant, Relation between c_p and c_v, Various non-flow processes like constant volume process, constant pressure process, Isothermal process, Adiabatic process, Polytropic process. Steam formation, Enthalpy, Specific volume, Internal energy and dryness fraction of steams, steam calorimeters.	5
Thermodynamic Cycle	Rankine cycle, Actual vapour cycle processes, Comparison of Rankine and Carnot cycles, Air standard cycles - Otto, Diesel, dual and Brayton cycles, Vapour compression refrigeration cycles.	8
Introduction to Mechanics of Solid:	Normal and shear Stress, strain, Hookes' law, Poisson's ratio, elastic constants and their relationship, stress-strain diagram for ductile and brittle materials, factor of safety. Basic Numerical problems, temperature stresses, shear stress, complementary shear stress, shear strain.	8
Compound Stresses and Strains	State of stress at a point, oblique stress, simple tension, pure shear, general two dimensional stress system, principal planes, principal stresses and strains, maximum shear stress.	8
Bending Stress and Torsion	Pure bending, moment of inertia, section modulus, bending stresses, combined bending and direct stress, beam of uniform strength, middle third and middle quarter rules for rectangular and circular sections, Circular shafts, torsional shear stress, strain energy in torsion, shafts under varying torque, compound shafts, combined bending and twisting.	8
	Total No. of Hours	**45**
Textbooks	1. R S Khurmi, "Engineering Mechanics". 2. P K Nag "Engineering Thermodynamics".	
References	1. Van Wylen G.J. & Sonnlog R.E.: Fundamentals of classical thermodynamics, John Wiley & Sons,Inc. NY. 2. Wark Wenneth : Thermodynamics (2nd edition), Mc Graw Hill book Co. NY. 3. Holman, J.P.: Thermodynamics, MC Graw Hill book Co. NY. 4. Yadav R.: Thermodynamics and Heat Engines, Vol I & II (Sl Edition) Central Publishing House Allahabad. 5. Yadav R.: Steam & Gas Turbines. 6. Kshitish Chandra Pal: Heat Power, Orient Longman Limited, 17, Chittranjan Avenue, Calcutta. 7. S. Rao, B.B. Parulekar, 'Energy Technology', Khanna Pub., New Delhi. 8. G. H. Ryder: "Strength of Materials". 9. F. L. Singer: "Strength of Materials". 10. Timoshenko: "Strength of Materials". 11. Beer, Johnson, Statics".	

SET/CS/BT/C105 C PROGRAMMING		
Course Objective	1. To introduce the fundamental concepts of programming using the C language. 2. To develop problem-solving skills and logical thinking through the creation and execution of C programs. 3. To teach the use of control statements, functions, pointers, arrays, and data structures in C. 4. To provide hands-on experience with file input/output operations in C. 5. To prepare students for more advanced programming courses and practical applications in various fields.	
Course Outcome	1. Develop programs in C programming language. 2. Analyze the problem and find appropriate solution 3. Evaluate the correctness of the developed solution. 4. Develop basic and advanced level applications using C programming language.	
Module Name	**Content**	**No. of Hrs.**
Introduction	Introduction, The C character set, Constants, Variables, Identifiers, Keywords, Data types, Declarations, The First C Program, Compilation and Execution.	6
Operators and Expressions	Arithmetic, Relational, Equality, Logical, Unary, Conditional, Bitwise, Assignment, Comma and Size of operator. Type Conversion and Typecasting.	6
Control Statements	if, if-else, while, do-while, for loop, nested loops, switch, break, continue and goto statements.	5
Functions & Pointers	Defining and accessing functions, Function prototype, Passing arguments, Recursion, Use of library functions. Introduction to pointers, Declarations, Passing to a function, Operations on pointers, Dynamic memory allocation, Array of pointers.	11
Arrays	Single and Multi-dimensional arrays, Row major and Column major form of an array, Character strings and arrays.	4
Storage classes	Automatic, Register, Static and External storage class.	4
Structures and Unions	Basics of structures, Structures and functions, Arrays of Structures, Pointers to structures, Self-referential structures, Unions.	4
File Input/output	Opening a File, Reading from a file, closing the file, Writing to a file.	4
Total No. of Hours		**44**
Textbooks	1. E. Balagurusamy, “Programming in ANSI C”	
References	1. Byron S. Gottfried, “Programming With C” 2. Yashwant Kanitker, “LET US C” 3. B. W. Kernighan and D. M. Ritchie, “The C Programming Language” 4. B. W. Kernighan, “The Practice of Programming”, Addison-Wesley, 1999. 5. C. L. Tondo and S. E. Gimpel, “The C Answer Book”, (2/e), Prentice Hall, 1988.	

SET/SH/BT/C108	CHEMISTRY LAB	
Module Name	**Content**	**No. of Hrs.**
	1. To determine the percentage of available chlorine in the supplied sample of bleaching powder. 2. To determine the ferrous content in the supplied sample of iron ore by titrimetric analysis against standard K2Cr2O7 solution using K3Fe(CN)6 as external indicator. 3. To determine the chloride content in supplied water sample using Mohr's method. 4. To determine the constituents and amount of alkalinity of the supplied water sample. 5. To determine the temporary and permanent hardness of water sample by complexometry. 6. To find chemical oxygen demand of a waste water sample using Potassium Dichromate. 7. To determine iron concentration in the sample of water by Spectrophotometric method. 8. To determine the molecular weight of a polystyrene sample by using viscometric method. 9. To determine pH of a solution by using digital pH meter and titration of such a solution pH metrically. 10. Analysis of a coal sample by proximate analysis method.	3 x 10
Total No. of Hours		**30**

SET/CS/BT/C109	C PROGRAMMING LAB
Course Objective:	1. To make the student learn a programming language. 2. To learn problem solving techniques. 3. To teach the student to write programs in C and to solve the problems.
Course Outcome:	1. After Completion of this course the student would be able to 2. Read, understand and trace the execution of programs written in C language. 3. Write the C code for a given algorithm. 4. Implement Programs with pointers and arrays, perform pointer arithmetic, and use the pre-processor. 5. Write programs that perform operations using derived data types.

Content	**No. of Hrs.**
This lab shall have minimum 25 programs in C. There shall be minimum two programs per module as taught in theory. Programming shall follow logic/algorithm and flowchart wherever applicable. Exercises shall also enhance analytical and debugging abilities.	2x16
Total No. of Hours	**32**

VAC-1 Understanding and Connecting Student with Environment*
As per University Proposal and Approval

SET/CS/SC/C110 Internet Technology Lab-I (Skill Enhancement Course)		
Course Objective:	1. To make the student learn a programming language. 2. To learn Microsoft office techniques. 3. To learn computer network and trending techniques	
Course Outcome:	1. After Completion of this course the student would be able to know about the office automation techniques and implement on day to day activities 2. Working with computer networking equipment and email.. 3. Implement Programs to design web development	
Module Name	**Content**	**No. of Hrs**
Module I	Working with Microsoft Office (Word, Excel, Power Point, Access)	10
Module II	Use of Search Engine and World Wide Web, Creation of email id and working with email, Use of FTP service	10
Module III	Basics of Cloud computing, Internet of things (IoT), Data Science, Artificial Intelligence, Block-Chain Technology, Client-Server Architecture, P2P Networks	10
	Besides these additional experiments can be included to give hands on experience to students.	
	Total Hours	**30**

Semester II

S. No.	Category	Course Code	Course Code and Title	L	T	P	Contact Hrs./Week	Credits
1	Basic Science/Multidisciplinary	SET/SH/BT/C201	Mathematics II	3	1	-	4	4
2		SET/SH/BT/C202	Physics	3	1	-	4	4
3	Core Basic Engineering Subjects	SET/EE/BT/C203	Basic Electrical Engineering	3	1	-	4	4
4		SET/EC/BT/C204	Basic Electronics	3	1	-	4	4
5		SET/IT/BT/C205	Fundamental of Information Technology	3	1		4	4
6	Core/Basic Subjects Based Labs	SET/SH/BT/C207	Physics Lab	-		1	3	1
7		SET/ME/BT/C208	Engineering Graphics and Workshop Practice			1	3	1
8	Life Skills and personality development	VAC-2	A Life Skills and personality Development*	2	-	-	2	2
9	Skill Course	SET/CS/SC/C210 or SET/CS/SC/C211	Internet Technology Lab-II or Basics of Python Lab (Skill Enhancement Course)	-	-	1	4	2
Total				17	5	3	32	26

* Common syllabus for all UC Courses of the University.

SET/SH/BT/C201	MATHEMATICS-II	
Course Objective:	1. To provide an in-depth understanding of multiple integrals and their applications in various fields. 2. To introduce the concepts and applications of Fourier series and integral transforms. 3. To familiarize students with the principles and techniques of probability and statistics. 4. To develop analytical skills for solving complex problems involving integration and statistical analysis. 5. To prepare students for advanced studies and research in mathematics, engineering, and physical sciences.	
Course Outcome:	1. Students will be able to evaluate double and triple integrals and apply them to compute area, volume, centre of gravity, and moments of inertia. 2. Students will understand and apply the concepts of Fourier series to represent periodic functions and solve related problems. 3. Students will gain proficiency in using Laplace transforms to solve linear differential equations and understand their applications in various contexts. 4. Students will learn the fundamental concepts of probability and statistics, including different types of distributions and their properties. 5. Students will be able to perform statistical analyses, including correlation, regression, and conditional probability, and apply Bayes theorem in problem-solving.	
Module Name	**Content**	**No. of Hrs.**
Multiple Integral	Evaluation of definite integral; double and triple integrals; change of order of integration. Change of variables, application to area, volume, centre of gravity, moment of inertia and product of inertia. Gamma and Beta functions, Dirichlet's integral and its application.	**12**
Fourier Series	Periodic functions, Fourier series of functions with period 2n, change of interval, half range sine and cosine series	**6**
Integral Transform	Laplace transforms, existence theorem, Laplace transform derivatives, inverse Laplace transform, application to solve linear differential equations, unit step function, Dirac delta function, Laplace transforms of periodic functions. Application of Laplace transforms. Definitions of Fourier transform and its simple applications	**14**
Probability and Statistics	Random variables. Uniform, normal, exponential, Poisson and binomial distributions. Mean, median, mode and standard deviation, Correlation and regression, Conditional probability and Bayes theorem	**12**
	Total No. of Hrs.	**44**
Textbooks	1. R. K. Jain and S. R. K. Iyengar "Advanced Engineering Mathematics", Narosa Publications, 2. B. S. Grewal, "Higher Engineering Mathematics", Khanna Publishers, 3. H K Das, "Advanced Engineering Mathematics", S Chand, 4. Erwin Kreyszig, "Advanced Engineering Mathematics".	

SET/SH/BT/C202	PHYSICS	
Course Objective	**1.** To introduce the student to the basic of wave optics, lasers, and demonstrate their applications in technology. **2.** To make students aware about quantum physics phenomena. **3.** Give the beginning student an appreciation of recent developments in materials science & engineering within the framework of this class. **4.** To review physics in the context of materials science & engineering. **5.** Give an introduction to the relation between processing, structure, and physical properties. **6.** To make the students aware about Electromagnetic wave fundamentals.	
Course Outcome	1. Demonstrate interference, diffraction and polarization of light and explain the working principle of Lasers. 2. Student will understand quantum mechanical aspects of physics. 3. Enable to explain the phenomenon of crystal structure and crystallographic, qualitatively description of X-ray diffraction and its general physical properties, as well as possible applications. 4. Students will understand the phenomenon of defects in solids and their physical properties, band theory of solids and classification of energy bands, electric and magnetic properties of solids and able to explain qualitative idea of superconductivity in materials. 5. This will enable the students to learn physical concepts associated with electromagnetic radiation and devices. 6. Use Maxwell's equations to describe propagation of EM waves in a medium.	
Module Name	**Content**	**No. of Hrs.**
Optics	Interference: Coherent Sources, Conditions of Interference, Fresnel's Biprism Experiment, Interference in Thin Films, Newton's Rings; Single and n-Slit Diffraction, Diffraction Grating, Raleigh's Criterion of Resolution, Resolving Power of Telescope, Phenomenon of Double Refraction, Ordinary and Extra-ordinary Rays, Nicol Prism, Circularly and Elliptically Polarized Light, Fresnel Theory, Optical Activity, Specific Rotation; Laser: Principle of Laser Action, Einstein's Coefficients, Construction and Working of He-Ne and Ruby Laser, Applications of Laser.	15
Origin of Quantum Mechanics and its Applications	Black body radiation, Planck's Radiation Law, Wave Particle Duality, deBroglie hypothesis, Photoelectric effect, Wave Function and its Normalization, Born Interpretation, Schrodinger equation, Particle in a Box, Potential Step ($E < Vo$), Tunnelling effect (Qualitative idea).	10
Basics Material Science	Introduction to crystal structure of materials, Miller indices for crystallographic planes and directions. Diffraction of X-Rays, Bragg's Law, Determination of crystal structure using X-rays Diffraction and its applications. Defects in solids: point, line and planar defects and their effect on properties of materials. Band theory of solids, conductors, semiconductors and insulators, metals. Fermi Level. Magnetism: dipole moments, paramagnetism, Curie's law, magnetization and hysteresis, Ferromagnetism and Anti- Ferromagnetism. Ferro electricity and Piezoelectricity. Superconductivity in materials.	15
Electromagnetics	Ampere's Law and Displacement Current, Maxwell's Equations in Integral and Differential Forms, Electromagnetic Wave Propagation in Free Space and Conducting Media, Poynting Theorem.	8
	Total No. of Hours	**48**
Textbooks	1. Gaur, Gupta, "Engineering Physics" 2. Callister W.D., "Materials Science and Engineering: An introduction", 6th Edition, John Wiley & Sons Inc., New York 2002.	
References	1. J. R. Taylor, C.D. Zafiratos and M. A. Dubson, Modern Physics for Scientists and Engineers,2nd Pearson 2. Arthur Beiser, Concepts of Modern Physics, 6th Ed., TMH, (2009) 3. D.J. Griffith : Electrodynamics 4. Charles Kittel, Introduction to Solid State Physics, 5. S.O. Pillai, Solid State Physics, 6. Ajoy Ghatak- Optics	

SET/EE/BT/C203 BASIC ELECTRICAL ENGINEERING		
Course Objective	1. To impart basic knowledge of electrical quantities and provide working knowledge for the analysis of DC and AC circuits. 2. To understand the construction and working principle of DC and AC machines. 3. To understand the construction and working principle of various instruments. 4. To understand the construction and working principle of 3- phase supply system.	
Course Outcome	1. Understand the basic electric and magnetic circuits. 2. Analyze DC and AC circuits. 3. Interpret the construction and working of different types of electrical machines and instruments. 4. Analyze basic electrical components and circuits.	
Module Name	**Content**	**No. of Hrs.**
DC Networks	Concepts of linear, nonlinear, active, passive, unilateral and bilateral elements; Ideal and practical voltage & current sources, conversion from one from the other; Kirchhoff's laws, statements; Mesh Analysis; Nodal Analysis; Delta-Star & Star-Delta conversion; Superposition principle; Thevenin's theorem, statement, advantages in case of complex networks; explanation & illustration with examples; Norton's theorem, Maximum power transfer theorem, Reciprocity Theorem and its application.	10
Single Phase AC Circuits	Generation of single phase AC voltage and determination of average (mean) and RMS (effective) values of voltage and current with special reference to sinusoidal waveforms; Form factor and peak factor for various waves; Representation of sinusoidal time varying quantities as phasors; concepts of reactance, impedance and their representation in complex forms using j operator; Steady state analysis of series R-L-C circuit & its phasor diagram; Concept of power & power factor; Concept of admittance, susceptance in parallel circuits; Analysis of series parallel circuits & phasor diagrams; Resonance in series and parallel circuits.	10
Three Phase Circuits	Generation of 3-phase balanced sinusoidal voltage; star & delta connections; line & phase quantities (current & voltage); Solution of 3-phase star/delta circuits with balanced supply voltage and balanced load; phasor diagram; 3-phase, 4-wire circuits; Measurement of three phase power by two wattmeter method; phasor diagram with balanced load and determination of load power factor from wattmeter readings.	6
Transformers and Rotating Machines	Transformers: Constructional features and principle of operation, concept of ideal transformer under no load & loaded conditions and its equivalent circuit; Practical transformer rating & its equivalent circuit; Autotransformer – principle of operation & relative advantages & disadvantages; Rotating Machine: construction features (stator, rotor & air gap), conditions for production of steady electromagnetic torque; Three phase Induction motor: constructional features and operation; DC Machines: construction features, EMF and Torque expression, Classification of DC motors and generators; Stepper motor.	12
Measuring Instruments	DC PMMC instruments – constructional feature and principle of operation; Moving iron meters construction and principle of operation; Dynamometer type wattmeter; Induction type energy meter construction & principle of operation.	6
	Total No. of Hours	**44**
Textbooks	1. I.J. Nagrath, "Basic Electrical Engineering," Tata Mc. Graw Hill.	
References	1. A. E. Fitgerald, D.E., Higginbotham and A Grabel, "Basic Electrical Engineering", Mc Graw Hill. 2. Rizzoni, Principles and Applications of Electrical Engineering, TMH. 3. V. Del Toro. "Principles of electrical Engineering, "Prentice hall. 4. W.H. Hayt & J.E. Kemmerly," Engineering circuit Analysis, "Mc Graw Hill. 5. H. Cotton, "Advanced Electrical Technology" Wheeler Publishing.	

SET/EC/BT/C204	Basic Electronics	
Course Objective	To familiarize the students with electronics field. To introduce semiconductor fundamentals, electronic devices, and elementary electronic circuits. To familiarize students with digital logics and gates.	
Course Outcome	1. Understand the working and current voltage characteristics of semiconductor devices e.g. diodes and transistor. 2. Perform dc analysis of amplifier circuits. 3. Design basic OP AMP circuits. 4. Understand and use basic digital electronic concepts.	
Module Name	**Content**	**No. of Hrs.**
Semiconductor Diodes	Semiconductor materials- intrinsic and extrinsic types, Ideal Diode as a switch, Terminal characteristics, and equivalent circuit of PN diode: p-n junction under open circuit condition, p-n junction under forward bias and reverse bias conditions, p-n junction in breakdown region; Zener diode and basic voltage regulator using Zener diode; Rectifier Circuits, Clipping and Clamping circuits; LED, Photo Diode.	10
Bipolar Junction Transistors	Physical structure, physical operation and current-voltage characteristics of NPN transistor; Use of Voltage-dependent Current source as a Voltage amplifier; Transistor as an amplifier: Characteristics of CE amplifier; Active region operation of transistor; D.C. analysis of Common Emitter Amplifier: load line analysis; Transistor as a switch: cut-off and saturation modes.	10
Field Effect Transistor	Enhancement-type MOSFET: structure and physical operation, currentvoltage characteristics; MOSFET as a Switch, MOSFET as a Voltagedependent Current source and Common Source Amplifier.	8
Operation Amplifier	Ideal Op-amp; Properties of the ideal Operational Amplifier; op-amp application circuits (assuming ideal op amp): inverting amplifier, non inverting amplifier, weighted summer, integrator, and differentiator.	6
Digital Logic and Gates	Binary, octal, and hexadecimal number systems; Methods of base conversions; Binary, octal, and hexadecimal arithmetic; Representation of signed numbers; Basic logic operations and logic gates; MOSFET Switch Implementation of Logic Gates, e.g., Inverter, NAND, NOR. Basic postulates and fundamental theorems of Boolean algebra.	8
	Total No. of Hours	**42**
Textbooks	1. Agarwal, Anant; Lang, Jeffrey H, "Foundations of Analog and Digital Electronic Circuits", Elsevier Science & Technology Books.	
References	1. V. Del Toro, Principles of Electrical Engineering, PHI. 2. Rizzoni, Principles and Applications of Electrical Engineering, TMH. 3. Malvino, Electronic Principles. 4. R.L.Boylestad & L.Nashelsky, Electronics Devices & Circuit Theory, PHI. 5. Sedra, Smith, "Microelectronic Circuits", Oxford University Press.	

SET/IT/BT/C205	Fundamentals of Information Technology	
Course Objective	1. Take on significant positions In various IT work 2. Collaborate in diverse team environments 3. Contributions in the field of IT 4. Work effectively in the IT field to make a positive contribution to society	
Course Outcome	1. Develop information technology solutions by evaluating user requirements in the systems development environment. 2. Apply knowledge of IT requirements for technology solutions in cutting edges applications. 3. Analyze a problem and identify and define the computing requirements for the appropriate solutions. 4. Create, select and apply appropriate techniques, resources, and modern engineering and IT tools.	
Module Name	**Content**	**No. of Hrs.**
Introduction	Definition of Electronic Computer, Generations, Classification of Computers, Computer Hardware and Basic Computer Organization: CPU- ALU, CU; RAM/ROM, Various I/O devices, Peripherals, Storage Media	4
Computer Languages	Binary, Hexadecimal Number System; Basic Binary Logic Operations; Binary Addition and Subtraction; Generation of Languages, Assembly Language, High level language; Translators, Interpreters, Compilers, Compilers; Flow Charts, Dataflow Diagram,	6
OS & Office	Software- System and Application Software; Elementary Concepts in Operating System; Textual Vs GUI, Introduction to DOS, MS Windows, UNIX/Linux	4
Computer Networks	Elements of Communication system; Brief Introduction to Computer Networks- Introduction of LAN and WAN. Network Topologies, Client-server Architecture, IoT, Cloud Computing	6
Internet	Internet & World Wide Web, Hypertext Markup Language, DHTML, Python, WWW, Gopher, FTP, Telnet, Web Browsers, Net Surfing, Search Engines, Email; Introduction to Web Development, Static and Dynamic Pages	6
IT Application and Multi media	Basic Awareness of NICNET and ERNET; E Commerce, E governance; Brief Introduction to Different Formats of Image, Audio, Video	6
Information Concepts & Processing	Definitions of Information, Need of information, quality of information, value of information, concept of information, Entropy category and Level of information in Business Organization, Data Concepts and Data Processing, Data Science, Data Representation, Application of IT to E-commerce, Electronic Governance, Multimedia, Entertainment, Introduction to Information System.	8
	Total No. of Hours	**40**
Textbooks	1. Sinha, Sinha, “Computer Fundamentals”, 2. Yadav R. P. , “Information Technology”	
References	1. D S Yadav, “Foundations of IT”, New Age, Delhi 2. Rajaraman, “Introduction to Computers”, PHI 3. Peter Nortans “Introduction to Computers”, TMH. 4. Patterson D.A. & Hennessy J.L., "Computer Organization and Design", Morgan Kaufmann Publishers	

SET/SH/BT/C207	PHYSICS LAB	
Course Objective	**1.** To introduce students to the experimental verification of fundamental physical theories. **2.** To familiarize students with the principles and applications of spectroscopy, lasers, and semiconductor experiments. **3.** To provide hands-on experience in conducting laboratory experiments related to key physical concepts. **4.** To demonstrate the practical applications of various physical laws and phenomena. **5.** To develop students' skills in data analysis, experimental techniques, and scientific reporting.	
Course Outcome	1. After Demonstration the student will able to perform the experiment and learn about the practical knowledge of various theory part. 2. Student will enable to find the refractive index of material, wavelength of monochromatic source of light. 3. Enable to find the efficiency of electric kettle, band gap of materials, behaviour of semiconductor, charge density and hysteresis curve in ferromagnetic materials	
Sr. No.	**Experiments**	**No. of Hrs.**
1.	To determine refractive index of glass and liquid using spectrometer.	1x2
2.	To determine the wavelength of spectral lines using plane diffraction grating (Use Hg source).	1x2
3.	To determine the wavelength of sodium light by Newton's Ring method.	1x2
4.	To measure an accessible (Horizontal and vertical) height using sextant.	1x2
5.	Determination of wavelength of He-Ne laser using single slit /N slit diffraction pattern.	1x2
6.	To study the photoelectric effect and determine the value of Planck's constant.	1x2
7.	To determine the heating efficiency of an electric kettle with varying voltage.	1x2
8.	To Determine the wavelength of the semiconductor diode laser.	1x2
9.	Measurement of forward/reverse saturation current in p-n-junction diode at various temperatures and to find the approximate value of energy gap.	1x2
10.	To study the Hall effect and determine Hall coefficient, carrier density and mobility of a given semiconductor material.	1x2
11.	To draw hysteresis curve of a given sample of ferromagnetic material and from this to determine magnetic susceptibility	1x2
12.	Measurement of e/m of electron e/m- Thomson's Experiment	1x2
13.	To verify Ohm's law.	1x2
14.	Conversion of Galvanometer into Voltmeter and Ammeter.	1x2
15.	To determine the unknown resistance by a post office box.	1x2
	Total No. of Hours	**30**

References	1. Practical Physics, C.L. Arora, S. Chand & Co. 2. Engineering Practical Physics, S.Panigrahi & B.Mallick, 2015, Cengage Learning India Pvt. Ltd. 3. Advanced Practical Physics for students, B.L. Flint & H.T. Worsnop, 1971, Asia Publishing House. 4. Advanced level Physics Practicals, Michael Nelson and Jon M. Ogborn, 4th Edition, reprinted 1985, Heinemann Educational Publishers. 5. A Text Book of Practical Physics, Indu Prakash and Ramakrishna, 11th Edition, 2011, Kitab Mahal, New Delhi.

SET/ME/BT/C208	Engineering Graphics and Workshop Practice	
Course Objective	1. To introduce students to the fundamental principles of engineering graphics and familiarize them with technical drawing instruments and standards. 2. To develop the ability to create and interpret technical drawings, including projections of points, lines, planes, and solids. 3. To impart practical skills in carpentry, fitting, blacksmithing, welding, and machining through hands-on laboratory experiences. 4. To enhance students' understanding of orthographic projection and its application in creating detailed drawings of machine components. 5. To provide a comprehensive understanding of the operations and applications of various workshop machines such as lathe, shaper, milling, and drilling machines.	
Course Outcome	1. Students will be proficient in using drawing instruments and applying current Indian Standard Code of Practice for Engineering Drawing. 2. Students will be able to accurately project points, lines, and planes in different coordinates and understand the concepts of true lengths and traces. 3. Students will develop the skills to project polyhedral and solids of revolution, including projections with axes inclined to reference planes. 4. Students will gain the ability to convert pictorial views into orthographic projections and create detailed technical drawings of machine components. 5. Students will acquire hands-on experience in various workshop practices, including carpentry, fitting, blacksmithing, welding, and machining, enabling them to perform basic manufacturing and assembly tasks.	
Module Name	**Content**	**No. of Hrs.**
Introduction to Engineering Graphics & Projection of Points	Drawing instruments and their use, Different types of lines, Lettering & dimensioning Familiarization with current Indian Standard Code of Practice for Engineering Drawing. Scales, Plain scales, Diagonal scales, Vernier scales. First angle and third angle projections Projection of points in different coordinates, Projections of lines inclined to one of the reference planes.	08
Projections of lines and planes	Projections of lines inclined to both the planes, True lengths of the lines and their angles of inclination with the reference planes, Traces of lines. Projection of plane lamina of geometric shapes inclined to one of the reference planes, inclined to both the planes, Traces of planes. Projections on auxiliary planes.	08
Projections of polyhedral and solids	Projections of polyhedral and solids of revolution, projection of solids with axis parallel to one of the planes and parallel or perpendicular to the other plane, Projections with the axis inclined to one of the planes.	08
Orthographic Projection	Concept of orthographic projection, Rules of Drawing orthographic projection, Conversion of pictorial views into orthographic projection, Drawing of orthographic projection of Machine components.	08
Carpentry, Fitting and Black smithy	Minimum two experiments from Carpentry, Fitting and Black smithy. And Development of jobs carried out and soldering, Black Smithy, House Wiring, Foundry (Molding only), Plumbing.	08
Welding & Machining	Practice of minimum two experiments of welding joints. Overview of Lathe, Shaper, Milling and Drilling machine. Perform one job on each machine.	08
	Total No. of Hours	**48**
Textbooks	4. Bhatt N. D, Elementary Engineering Drawing, Charotar Publishing House, Anand, 2002. 5. Elements Of Workshop Technology Vol-1 by Hazra Chaudhary	

References	1. Narayana K L & Kannaiah P, Engineering Graphics, Tata McGraw Hill, New Delhi, 1992. 2. Luzadder W J, Fundamentals of Engineering Drawing, Prentice Hall of India, New Delhi, 2001. 3. Thomas E French & Charkes J V, Engineering Drawing & Graphing Technology, McGraw Hill Book Co, New York, 1993. 4. Venugopal K, Engineering Drawing & Graphics, New Age International Pvt. Ltd., New Delhi, 1994. 5. Workshop Technology, Raghubanshi.

VAC-2 A Life Skills and personality Development*
As per University Proposal and Approval

SET/CS/SC/C210 Internet Technology Lab-II (Skill Enhancement Course)		
Course Objective:	1. To make the student learn about web development. 2. To learn about static and dynamic web pages.	
Course Outcome:	1. After Completion of this course the student would be able to know about the web development . 2. Working with HTML/CSS/Javascript for designing web pages.	
Module Name	**Content**	**No. of Hrs**
Module I	Introduction to Web Development	10
Module II	Creation of Static Web Pages using HTML/CSS	10
Module III	Creation of Page Using Java Script	10
	Besides these additional experiments can be included to give hands on experience to students.	
	Total Hours	**30**

SET/CS/SC/C211 Basics of Python Lab (Skill Enhancement Course)		
Course Objective:	1. To make the student learn about Python programming language. 2. To develop basic programs using primitive data structures.	
Course Outcome:	After Completion of this course the student would be able to know about the basic Python programming.	
Module Name	**Content**	**No. of Hrs**
Module I	Install Python and write your first program	5
Module II	Describe the basics of the Python programming language	10
Module III	Use variables to store, retrieve and calculate information, Utilize core programming tools such as functions and loops	15
	Besides these additional experiments can be included to give hands on experience to students.	
	Total Hours	**30**

Semester III

S. No.	Category	Course Code	Course Code and Title	L	T	P	Contact Hrs./Week	Credits
1	Basic Science/Multidisciplinary	SET/AH/BT/C301	Mathematics III	3	1	-	4	4
2	Core Subjects	SET/CS/BT/C302	Computer Based Numerical & Statistical Techniques	3	1	-	4	4
3		SET/CS/BT/C304	Data Structures Using C	3	1	-	4	4
4		SET/CS/BT/C305	Discrete Structures	3	1	-	4	4
5	Interdisciplinary Subject	SET/EC/BT/C303	Digital Electronics	3	1		4	4
6	Core Subjects Based Labs	SET/CS/BT/C306	Computer Based Numerical & Statistical Techniques lab	-		1	3	1
7		SET/CS/BT/C307	Digital Electronics Lab			1	3	1
8	Extracurricular Courses/CC	**AMDSC-2**	***Additional Multidisciplinary Skill course (AMSC): Any one of the following** **1. Nursery training course** **2. Basic Yoga practices** **3.Physical education/sports management** **4. Folk and culture** **5. Indian traditional music**	2	-	-	2	2
9	Skill Course	SET/CS/SC/C308	Data Structures Using C Lab	-	-	1	4	2
Total				17	5	3	32	26

***** Compulsory for all U.G. students, to be prepared by University.

SET/CS/BT/C301 MATHEMATICS- III		
Course Objective	1. To introduce students to the fundamental concepts of ordinary and partial differential equations and their applications in engineering. 2. To provide students with various solution methods for ordinary differential equations, including first-order and linear equations of nth order with constant coefficients. 3. To familiarize students with partial differential equations of various types (parabolic, elliptic, hyperbolic) and their classifications. 4. To teach students numerical methods for solving differential equations and numerical integration techniques. 5. To introduce students to the theory of complex variables, including analytic functions, Cauchy's integral theorem, and residue theorem..	
Course Outcome	1. Students will be able to classify and solve ordinary differential equations of various orders and types, including homogeneous and non-homogeneous equations. 2. Students will gain proficiency in solving linear partial differential equations with constant coefficients and understand the physical significance of solutions in engineering problems. 3. Students will develop the ability to use numerical methods to approximate solutions of differential equations and perform numerical integration using trapezoidal and Simpson's rules. 4. Students will demonstrate an understanding of complex variables, including analytic functions, Cauchy-Riemann equations, and Taylor/Laurent series expansions. 5. Students will be able to apply complex variable theory to solve engineering problems, including evaluating integrals using the residue theorem and analyzing functions with poles and singularities.	
Module Name	**Content**	**No. of Hrs.**
Ordinary Differential Equations	Introduction to order, degree and arbitrary constants, solution methods for differential equations of first order , linear differential equations of n^{th} order with constant coefficient, complimentary functions and particular integrals, Homogeneous differential equations, Cauchy's and Euler's equations, Method of variation of parameters, equations of the form $y'' = f(y)$, applications to engineering problems.	**12**
Partial Differential Equations	Linear PDE with constant coefficients of 2nd order and their classifications, Initial and boundary value problems, PDE of parabolic, elliptic and hyperbolic type. Separation of variables method for solving PDE, heat equations, wave equations and Laplace equations.	**10**
Numerical Methods	Direct and iterative methods to solve of linear algebraic equations, numerical integration, integration by trapezoidal and Simpson‟s rules.	**08**
Complex Variables	Analytic functions; Cauchy-Riemann equations; Harmonic functions, Cauchy‟s integral theorem and integral formula; sequences, series, convergence tests, Taylor and Laurent series, poles and singularity of zeros, residue theorem.	**12**
	Total No. of Hrs.	**42**

Textbooks	1. R. K. Jain and S. R. K. Iyengar "Advanced Engineering Mathematics", Narosa Publications, 2. B. S. Grewal, "Higher Engineering Mathematics", Khanna Publishers, 3. H K Das, "Advanced Engineering Mathematics", S Chand, 4. Erwin Kreyszig, "Advanced Engineering Mathematics".

SET/CS/BT/C302 COMPUTER BASED NUMERICAL & STATISTICAL TECHNIQUES		
Course Objective	1. To familiarize students with errors that occur in numerical computations and provide them with the necessary mathematical preliminaries for error analysis. 2. To introduce students to various methods for solving algebraic and transcendental equations, including bisection method, Newton's-Raphson method, and interpolation techniques. 3. To teach students about interpolation methods, curve fitting procedures, cubic spline interpolation, and approximation of functions. 4. To provide students with numerical techniques for integration and differentiation, including trapezoidal rule, Simpson's rules, and Gaussian formula. 5. To introduce students to statistical computation techniques, including regression analysis, least square fit, and statistical control methods.	
Course Outcome	1. Recognize the error in the number generated by the solution. 2. Compute solution of algebraic and transcendental equation by numerical methods like Bisection method and Newton Rapshon method. 3. Apply method of interpolation and extrapolation for prediction. 4. Recognize elements and variable in statistics and summarize qualitative and quantitative data. 5. Calculate mean, median and mode for individual series. 6. Outline properties of correlation and compute Karl-Pearson‟s coefficient of correlation.	
Module Name	**Content**	**No. of Hrs.**
Errors in numerical computations	Errors in numerical computations, mathematical preliminaries, errors and their analysis, machine computations, computer software	6
Algebraic & Transcendental Equation	Bisection method, iteration method, method of false position, rate of convergence, method for complex root, Muller‟s method, quotient difference method, Newton‟s-Raphson methods.	6
Interpolation	roduction, errors in polynomial interpolation, finite difference, decision of errors, Newton‟s formulae for interpolation, Guass, Stirling, Bessel's, Everett‟s formulae, interpolation by unevenly spaced points, Lagrange interpolation formula, divided difference, Newton‟s general interpolation, formula. Curve Fitting.	10
Cubic Spline & Approximation	Introduction, method of least square curve fitting procedures, fitting a straight line, curve fitting by sum of exponentials, data fitting with cubic splines, approximation of functions..	8
Numerical Integration & Differentiation	Introduction, numerical differentiation, numerical integration, trapezoidal rule, Simpson 1/3 rule, Simpson 3/8 rule, Booles and Weddles rule, Euler- Maclariaun formula, Gaussian formula, numerical evaluation of singular integrals.	6
Statistical Computation	Frequency chart, regression analysis, least square fit, linear & non-linear regression, multiple regression, statistical control methods.	6
	Total No. of Hours	**42**
Textbooks	1. Sashtry : Introductory Method of Numerical Analysis, PHI 2. 2. Balaguruswamy : Numerical Methods, TMH	
References	1. Jain, Iyengar, Jain : Numerical Methods for Scientific& Engg. Computation, New Age 2.Gerald & Wheatley : Applied Numerical Analysis, Addison Wesley	

SET/CS/BT/C304	Data Structures Using C	
Course Objective	1. To impart the basic concepts of data structures and algorithms. 2. To understand concepts about searching and sorting techniques. 3. To understand basic concepts about stacks, queues, lists, and trees, etc. 4. To understanding about writing algorithms and step by step approach in solving problems with the help of fundamental data structures	
Course Outcome	1. Ability to analyze algorithms and algorithm correctness. 2. Ability to summarize searching and sorting techniques 3. Ability to describe stack, queue and linked list operation. 4. Ability to have knowledge of tree and graphs concepts.	
Module Name	**Content**	**No. of Hrs.**
Elementary Data Organization	Introduction to Field, Record, Data and Elementary Data Organization, Basic operations, Algorithm Complexity and Time-Space trade-off.	6
Arrays and Linked list	Representation and Analysis, Single and Multidimensional Arrays, address calculation, application of arrays, String in C, Array as Parameters, Ordered List, Sparse Matrices, Representation and Implementation of Singly Linked Lists, Two-way Header List, Traversing and Searching of Linked List, Overflow and Underflow, Insertion and deletion to/from Linked Lists, Insertion and deletion Algorithms, Doubly linked list, Linked List in Array, Polynomial representation and addition, Garbage Collection and Compaction.	12
Stacks and Queues	Array Representation of Stack, Linked Representation of Stack, Operations Associated with Stacks, Application of stack: Conversion of Infix to Prefix and Postfix Expressions, Evaluation of postfix expression using stack. Array and linked representation and implementation of queues, Create, Add, Delete, Full and Empty, Circular queues, D-queues and Priority Queues.	8
Trees	General Trees Binary Trees, Binary tree representation, algebraic Expressions, Complete Binary Tree, Extended Binary Trees, Array and Linked Representation of Binary trees, Traversing Binary trees, Threaded Binary trees, Traversing, Threaded Binary trees, Huffman algorithm, Binary Search Tree, Insertion and Deletion in BST, AVL Trees, B-trees.	8
Searching and Sorting	Sequential search, binary search, comparison and analysis, Insertion Sort, Bubble Sorting, Quick Sort, Two Way Merge Sort, Heap Sort, Sorting on Different Keys, Complexity of Search Algorithm.	8
	Total No. of Hours	**42**
Textbooks	1. Seymour Lipschutz, "Data Structures", TMH.	
References	2. R. Kruse etal, "Data Structures and Program Design in C", Pearson Education Asia, Delhi2002. 3. A. M. Tenenbaum, "Data Structures using C & C++", Prentice-Hall of India Pvt. Ltd., New Delhi. Holman, J.P.: Thermodynamics, MC Graw Hill book Co. NY.	

SET/CS/BT/C305	Discrete Structure
Course Objective	1. To introduce students to the foundational concepts of set theory, including countable and uncountable sets, relations, functions, and mathematical induction. 2. To familiarize students with algebraic structures such as semigroups, monoids, groups, rings, and fields, along with their properties and applications. 3. To teach students about posets (partially ordered sets), Hasse diagrams, lattices, and their properties, including bounded and complemented lattices. 4. To provide students with a solid understanding of propositional and first-order logic, truth tables, logical operations, tautologies, contradictions, and algebra of propositions. 5. To enable students to apply permutation and combination techniques, recurrence relations, generating functions, and probabilistic methods in problem-solving contexts.
Course Outcome	1. Students will be able to demonstrate proficiency in manipulating sets, relations, and functions, and apply mathematical induction and the pigeonhole principle to prove statements and solve problems. 2. Students will gain a thorough understanding of algebraic structures, including groups, rings, and fields, and their properties, and be able to apply them in various mathematical contexts.

	3. Students will be able to construct Hasse diagrams for partially ordered sets, analyze lattice properties, and understand the concepts of bounded and complemented lattices. 4. Students will develop logical reasoning skills, including constructing truth tables, evaluating logical propositions, and understanding logical implications and equivalences. 5. Students will acquire the ability to solve problems related to permutations, combinations, recurrence relations, and generating functions, and apply probabilistic methods in counting problems.	
Module Name	**Content**	**No. of Hrs.**
Set Theory	Countable and uncountable sets, Venn Diagrams, proofs of some general identities on sets Relation: Definition, types of relation, composition of relations, Pictorial representation of m relation, equivalence relation, partial ordering relation, Type of functions, one to one, into and onto function, inverse function, composition of functions, recursively defined functions, mathematical induction (simple and strong), pigeonhole principle, prove by contradiction.	12
Algebraic Structures	Properties, Semi Groups, Monoid, Groups, Abelian group, properties of groups, Subgroup, cyclic groups, Cosets, factor group, Permutation groups, Normal subgroup, Homomorphism and isomorphism of Groups, Rings and Fields.	6
Posets, Hasse Diagram and Lattices	Introduction, ordered set, Hasse diagram of partially, ordered set, isomorphic ordered set, well ordered set, properties of Lattices, bounded and complemented lattices.	6
Propositional Logic	Proposition, First order logic, Basic logical operation, truth tables, tautologies, Contradictions, Algebra of Proposition, logical implications, logical equivalence, predicates, Universal and existential quantifiers.	10
Permutation & Combination	Recurrence Relation, Generating function., Permutation & Combination, Probabilistic Permutation & Combination.	8
	Total No. of Hours	**42**
Textbooks	1. Liptschutz, Seymour, " Discrete Mathematics", McGraw Hill. 3rd edition 2. Trembley, J.P & R. Manohar, "Discrete Mathematical Structure with Application to Computer Science", McGraw Hill, Reprint 2010	
References	1. Discrete Mathematics & its application with combinatory and graph theory, K.H.Rosen, TMH (6th ed). 2. C.L.Liu, „Discrete Mathematics" TMH.	

SET/EC/BT/C303	**DIGITAL ELECTRONICS**	
Course Objective	1. To revise and extend the basic knowledge of number system and logic gates. Simplification of the complex Boolean expression using K-map. 2. To understand the combinational and sequential logic circuits. 3. To get the basic knowledge of logic families and semiconductor memories.	
Course Outcomes	Student should be able to: 1. Describe and demonstrate the use of digital test equipments and its operating characteristics. 2. Identify and describe the combinational and sequential logic circuits. 3. Understand the different memory devices.	
Module Name	**Content**	**No. of Hrs.**
Introduction	Positional number system; Binary, octal and hexadecimal number systems; Methods of base conversions; Binary, octal and hexadecimal arithmetic; Representation of signed numbers; Fixed and floating point numbers. Definition and specification of combination logic; Truth table; Basic logic operation and logic gates; Binary coded decimal codes; Gray codes.	**6**
Boolean Algebra and Switching Functions	Basic postulates and fundamental theorems of Boolean algebra; Standard representation of logic functions - SOP and POS forms; Simplification of switching functions - K-map.	**4**

Logic Families	Diode, BJT and MOSFET as a switch. Introduction to different logic families; Electrical characteristics of logic gates – logic levels and noise margins, fan-out, propagation delay, transition time, power consumption and power-delay product; circuit description and operation; RTL; DTL, HTL, TTL and sub families, Brief idea of ECL, CMOS BI-CMOS.	**10**
Combinational Logic	Arithmetic modules: adders, subtractors and ALU; Design examples. Decoders, encoders, multiplexers and de-multiplexers; Parity circuits and comparators.	**6**
Sequential Logic	Basic sequential circuits- latches and flip-flops: SR-latch, D-latch, D flip-flop, JK flipflop, T flip-flop and their inter-conversions; Timing hazards and races; Meta-stability; Analysis of state machines using D flip-flops and JK flip-flops; Definition of state machines, synchronous sequential logic, shift register, counters-ripple and mod counters.	**12**
Semiconductor Memories	RAM, ROM, Content Addressable Memory, Charge Coupled Device Memory. PLAs, PALs and their applications; Sequential PLDs and their applications.	**4**
	Total No. of Hours	**42**

Textbooks	1. M. Morris Mano, "Digital Design".
References	1. Taub, Schilieng, "Digital Integrated Electronics". 2. Anad Kumar, "Digital principles and application". 3. **4.** John F Wakerly, "Digital Design: Principles and Practices", PrenticeThomas L. Floyd, "Digital Fundamentals", Pearson/ Prentice Hall. Hall. **5.** Ronald J. Tocci, "Digital Systems: Principles and Applications", Pearson/ Prentice Hall. **6.** Charles Roth, "Fundamentals of Logic Design", Jaico Publishing House.

SET/CS/BT/C306	**COMPUTER BASED NUMERICAL & STATISTICAL TECHNIQUES LAB**	
Course Objective	1. Develop skills in polynomial interpolation and error analysis. 2. Implement numerical methods for solving equations and analyze root convergence rates. 3. Apply Bessel's, Newton's, Stirling's, and Lagrange's methods for solving mathematical problems. 4. Implement the method of least square curve fitting. 5. Implement numerical differentiation using trapezoidal and Simpson 3/8 rules. 6. Analyze data using frequency chart, regression analysis, linear and polynomial fits.	
Course Outcome	1. Implement polynomial interpolation and analyze errors. 2. Apply numerical methods for solving algebraic and transcendental equations and analyze root convergence rates. 3. Apply various methods (Bessel's, Newton's, Stirling's, Lagrange's) to solve mathematical problems. 4. Implement the method of least square curve fitting. 5. Implement numerical differentiation using trapezoidal and Simpson 3/8 rules. 6. Analyze data using frequency chart, regression analysis, linear and polynomial fits.	
Module Name	**Content**	**No. of Hrs**
Module I	Write a Program to deduce errors involved in polynomial interpolation.	6
Module II	Write a Program for algebraic and transcendental equations using bisection, iterative, method of false position, also give rate of conversions of roots in tabular form for each of these methods.	6
Module III	Write a Program to implement Bessel‟s functions, Newton‟s, Stirling‟s, Lagrange‟s.	6
Module IV	Write a Program to implement method of least square curve fitting.	6
Module V	Write a Program to Implement numerical differential using trapezoidal, Simpson 3/8 rules.	6
Module VI	Write a Program to show frequency chart, regression analysis, linear square fit and polynomial fit.	6

	Total Hours	**36**

SET/CS/BT/C307 DIGITAL ELECTRONICS LAB		
Module Name	**Content**	**No. of Hrs.**
Experiments	1. Combinational Logic design using basic gates (Code Converters, Comparators). 2. Combinational Logic design using decoders and MUXs. 3. Arithmetic circuits - Half and full adders and subtractors. 4. Arithmetic circuits – design using adder ICs, BCD adder. 5. Flip flop circuit (RS latch, JK & master slave) using basic gates. 6. Asynchronous Counters. 7. Synchronous counters, Johnson & Ring counters. 8. Sequential Circuit designs (sequence detector circuit). 9. Transfer Characteristics , Measurement of Sinking and Sourcing currents etc. of TTL gates.	10x2
Model Sim Simulations	Writing and simulating programs for adder, decoder, multiplexer, de-multiplexer, up/down counter, universal shift register, Sequence Detector etc.	4x2
	Total No. of Hours	**28**

SET/CS/SC/C308 DATA STRUCTURES LAB		
Course Objective	1. Implement Stack, Queue, and Circular Queue using arrays and lists. 2. Implement Tree, Binary Tree, Tree Traversal, Binary Search Tree, and operations for insertion and deletion. 3. Implement popular Searching and Sorting Algorithms. 4. Develop problem-solving skills using data structures.	
Course Outcome	1. Develop proficiency in implementing data structures. 2. Gain practical experience in using arrays and lists to implement Stack, Queue, and Circular Queue. 3. Understand the concepts of Tree, Binary Tree, and Binary Search Tree, and learn to implement them efficiently. 4. Learn various searching and sorting algorithms and gain experience in implementing them.	
Module Name	**Content**	**No. of Hrs**
Module I	Array implementation of Stack, Queue, Circular Queue.	9
Module II	List implementation of Stack, Queue, Circular Queue.	9
Module III	Implementation of Tree, Binary Tree, Tree Traversal, Binary Search Tree, Insertion and Deletion in BST.	9
Module IV	Implementation of Searching and Sorting Algorithms.	9
	Total Hours	**36**

Semester IV

S. No.	Category	Course Code	Course Title	L	T	P	Contact Hrs./Week	Credits
1	Core Subjects	SET/CS/BT/C401	Object Oriented Programming using C++	3	1	-	4	4
2		SET/CS/BT/C402	Operating System	3	1	-	4	4
3		SET/CS/BT/C403	Computer Organization and Architecture	3	1	-	4	4
4		SET/CS/BT/C405	Theory of Computation	3	1	-	4	4
5	Interdisciplinary Subject	SET/CS/BT/C404	Data Communication and Computer Network	3	1		4	4
6	Core Subjects Based Labs	SET/CS/BT/C406	Object Oriented Programming using C++ Lab	-		1	3	1
7		SET/CS/BT/C407	Operating System Lab			1	3	1
8	Indian Knowledge System-I (IKS-2)	VAC-3	**Indian Knowledge System***	2	-	-	2	2
9	Skill Course	SET/CS/SC/C408	Mini Project (Based on C/C++)			1	4	2
		Total		17	5	3	32	26

* Compulsory for all U.G. students, to be prepared by University.

SET/CS/BT/C401 OBJECT ORIENTED PROGRAMMING USING C++		
Course Objective	1. Introduces Object Oriented Programming concepts using the C++ language. 2. Introduces the principles of data abstraction, inheritance and polymorphism 3. Introduces the principles of virtual functions and polymorphism. 4. Introduces handling formatted I/O and unformatted I/O. 5. Introduces exception handling.	
Course Outcome	1. Able to develop programs with reusability. 2. Develop programs for file handling. 3. Handle exceptions in programming. 4. Develop applications for a range of problems using object-oriented programming techniques.	
Module Name	**Content**	**No. of Hrs.**
Fundamental Concept	Object Oriented Programming Paradigm, Basic concepts of OOP, Objects, Classes, Data abstraction and Encapsulation, Inheritance, Polymorphism, Dynamic binding, Message passing, Applications of OOP. Introduction to C++, structure of C++ Program. Tokens, Keywords, Identifiers and Constants, Data Types, Declaration and Dynamic Initialization of Variables, Reference Variables, Operators in C++, Expressions and their types, Control Structure, Functions in C++, Function Overloading.	10
Classes, Objects and Constructors	C Structure Revisited, Specifying a class, Defining Member functions, Making an Outside function inline, nesting of member function, Private member function, arrays within class, Memory allocation for objects, static data members and member functions, Arrays of objects, Object as a function arguments, Friend function, Returning objects, pointers to members local classes. Constructors, Parameterized constructors, Multiple constructors in a class, constructors with default arguments, dynamic initialization of objects, copy constructor, dynamic constructors, constructing 2-D arrays, Destructors.	8
Inheritance	Derived class declaration, forms of inheritance, inheritance and member accessibility, constructors and destructors in derived classes, constructors invocation and data members initialization, overloaded member functions, types of inheritance.	8
Polymorphism	Defining operator overloading, Overloading Unary and Binary operators, Operator Overloading using friends, Manipulation of strings using operators, Rules for overloading operators. Need for virtual functions, pointer to derived class objects, array of pointers to base class objects, pure virtual functions, virtual destructor, Concatenation of strings.	6
Streams computation & Exception Handling	Predefined console streams, hierarchy of console stream classes, unformatted and formatted console I/O operations, manipulators, Files: Hierarchy of file stream classes, opening and closing, testing for errors, modes, pointers and their manipulators, sequential access. Exceptions and Exception handling mechanism, throwing and catching mechanism, Rethrowing an exception, list of exceptions, handling uncaught exceptions.	10
	Total No. of Hours	**42**
Textbooks	1. Balagurusamy "Object Oriented Programming with C++ " , TMH	
References	1. Budd,"Object Oriented Programming ", Addison Wesley. 2. Mastering C++ K.R Venugopal Rajkumar, TMH. 3. C++ Primer , "Lip man and Lajole", Addison Wesley.	

SET/CS/BT/C402	OPERATING SYSTEM	
Course Objective	1. Students will learn how Operating System is Important for Computer System. 2. To make aware of different types of Operating System and their services. 3. To learn different process scheduling algorithms and synchronization techniques to achieve better performance of a computer system. 4. To know virtual memory concepts. To learn secondary memory management	
Course Outcome	1. Understands the different services provided by Operating System at different level. 2. They learn real life applications of Operating System in every field. 3. Understands the use of different process scheduling algorithm and synchronization techniques to avoid deadlock. 4. They will learn different memory management techniques like paging, segmentation and demand paging etc.	
Module Name	**Content**	**No. of Hrs.**
Fundamental Concept	Operating System and Function, Evolution of Operating System, Batch, Interactive, Time Sharing and Real Time System, System Protection. Operating System Structure: System Components, System Structure, Operating System Services.	6
Concurrent Processes	Process Concept, Principle of Concurrency, Producer / Consumer Problem, Critical Section, Problem, Semaphores, Classical Problems in Concurrency, Inter Processes Communication, Process Generation, Process Scheduling. CPU Scheduling: Scheduling Concept, Performance Criteria Scheduling Algorithm, Evolution, Multiprocessor Scheduling.	8
Deadlock	System Model, Deadlock Characterization, Prevention, Avoidance and Detection, Recovery From Deadlock Combined Approach.	6
Memory Management	Basic Machine, Resident Monitor, Multiprogramming with Fixed Partition, Multiprogramming With Variable Partition, Multiple Base Register, Paging, Segmentation, Paged Segmentation, Virtual Memory Concept, Demand Paging, Performance, Paged Replaced Algorithm, Allocation of Frames, Thrashing, Cache Memory Organization, Impact on Performance.	8
I/O Management & Disk Scheduling	I/O Devices and The Organization of I/O Function, I/O Buffering, Disk I/O, Performance criteria in scheduling algorithms, Concept of FCFS scheduling algorithm, Concept of priority scheduling algorithm like SJF, Concept of non- preemptive and preemptive algorithms, Concept of roundrobin scheduling algorithm, , Concept of multi-level queues, feedback queues. Operating System Design Issues. File System: Basic File System, Access Control Verification, Logical File System, and Physical File System File-System Interface: File Concept, Access Methods, Directory Structure, Protection, and Consistency Semantics File-System Implementation: File-System Structure, Allocation Methods, Free-Space Management, Directory Implementation, Efficiency and Performance, Recovery.	10
Unix Operating System	Development Of Unix, Role & Function Of Kernel, System Calls, Elementary unix command & Shell Programming, Directory Structure, System Administration, ,Case study: UNIX Operating System	4
	Total No. of Hours	**42**
Text Books	1. Tannenbaum, "Operating System Design and Implementation", PHI.	

References	1. Milenekovie, "Operating System Concept", McGraw Hill. 2. Petersons, "Operating Systems", Addision Wesley. 3. Dietal, "An Introduction to Operating System", Addision Wesley. 4. Gary Nutt, "Operating System, A Modern Perspective", Addision Wesley.

SET/CS/BT/C403	**Computer Organization and Architecture**	
Course Objective	1. Discuss the basic concepts and structure of computers. 2. Understand concepts of register transfer logic and arithmetic operations. 3. Explain different types of addressing modes and memory organization. 4. Learn the different types of serial communication techniques. 5. Summarize the Instruction execution stages.	
Course Outcome	**1.** Understand the theory and architecture of central processing unit. **2.** Analyze some of the design issues in terms of speed, technology, cost, performance.\ **3.** Design a simple CPU with applying the theory concepts. **4.** Use appropriate tools to design verify and test the CPU architecture. **5.** Learn the concepts of parallel processing, pipelining and interprocessor communication. **6.** Understand the architecture and functionality of central processing unit. **7.** Exemplify in a better way the I/O and memory organization. **8.** Define different number systems, binary addition and subtraction, 2"s complement representation and operations with this representation.	
Module Name	**Content**	**No. of Hrs.**
Fundamental Concepts	CPU, memory, input-output subsystems, control unit. Instruction set architecture of a CPU–registers, instruction execution cycle, Performing of arithmetic or logical operations, Fetching a word from memory, storing a word in memory, Bus and Memory Transfers, Bus Architecture, Arithmetic Algorithms (addition, subtraction, Booth Multiplication), IEEE standard for Floating point numbers. General register organization, Register Transfers, Register Transfer Language.	10
Control Design	Execution of a complete instruction, Multiple-Bus organization, Hardwired Control, Micro programmed control, Microinstruction, address sequencing, Microinstruction with Next-address field, Prefetching Microinstruction.	8
Processor Design	Processor Organization: Stack organization, Addressing mode, Instruction format, Data transfer & manipulations, Program Control, Reduced Instruction Set Computer. Assembly levels programs, programming techniques such as looping, counting and indexing addressing modes, data transfer instructions, arithmetic and logic operations.	8
Input-Output Organization	I/O Interface, Modes of transfer, Interrupts & Interrupt handling, Direct Memory access, Input- Output processor, Serial Communication.	6
Memory Organization	Memory Hierarchy, Main Memory (RAM and ROM Chips), Auxiliary memory, Cache memory, Virtual Memory, Memory management hardware.	6
Pipelining and Parallel Processors	Basic concepts of pipelining, throughput and speedup, pipeline hazards. Introduction to parallel processors, Concurrent access to memory and cache coherency.	4
	Total No. of Hours	**44**
Textbooks	1. Morris Mano, "Digital Design" 2. Computer System Architecture, M. Mano(PHI)	
References	1. Computer Organization, Vravice, Zaky & Hamacher (TMH Publication) 2. Structured Computer Organization, Tannenbaum(PHI) 3. Computer Organization, Stallings(PHI).	

SET/CS/BT/C404 Theory Of Computation		
Course Objective	1. Understand basic properties of formal languages and formal grammars. 2. Understand basic properties of deterministic and nondeterministic finite automata. 3. Understand the relation between types of languages and types of finite automata. 4. Understanding the Context free languages and grammars, and also Normalising CFG. 5. Understanding the minimization of deterministic and nondeterministic finite automata. 6. Understand basic properties of Turing machines and computing with Turing machines. 7. Understand the concept of Pushdown automata and its application. 8. Know the concepts of tractability and decidability, the concepts of NP-completeness and NP-hard problem. 9. Understand the challenges for Theoretical Computer Science and its contribution to other **sciences.**	
Course Outcome	1. Knowledge Acquire a full understanding and mentality of Automata Theory as the basis of all computer science languages design - Have a clear understanding of the Automata theory concepts such as RE's, DFA's, NFA's, Turing machines, Grammar, halting problem, computability and complexity. 2. Cognitive skills (thinking and analysis). - Be able to design FAs, NFAs, Grammars, languages modelling, small compilers basics - Be able to design sample automata - Be able to minimize FA's and Grammars of Context Free Languages. 3. Professional Skill - Perceive the power and limitation of a computer - Solve the problems using formal language. Attitude- Develop a view on the importance of computational theory.	
Module Name	**Content**	**No. of Hrs.**
Finite Automata	Introduction to defining language, Kleene closures, Arithmetic expressions, defining grammar, Chomsky hierarchy, Finite Automata (FA), Transition graph, generalized transition graph. Nondeterministic finite Automata (NFA), Deterministic finite Automata (DFA), Construction of DFA from NFA and optimization, FA with output: Moore machine, Mealy machine and Equivalence, Applications and Limitation of FA, Arden Theorem, Pumping Lemma for regular expressions, Myhill - Nerode theorem.	12
Context free grammar	Ambiguity, Simplification of CFGs, Normal forms for CFGs, Pumping lemma for CFLs, Decidability of CFGs, Ambiguous to Unambiguous CFG.	8
Push Down Automata	Description and definition, Working of PDA, Acceptance of a string by PDA, PDA and CFG, Introduction to auxiliary PDA and Two stack PDA.	10
Turing Machines	Basic model, definition and representation, Language acceptance by TM, TM and Type – 0 grammar, Halting problem of TM, Modifications in TM, Universal TM, Properties of recursive and recursively enumerable languages, unsolvable decision problem, undecidability of Post correspondence problem, Church‟s Thesis, Recursive function theory, Godel Numbering.	10
	Total No. of Hours	**42**
Text Books	1. K.L.P. Mishra and N. Chandrasekaran, “Theory of Computer Science (Automata, Languages and Computation)”, PHI	
References	1. Hopcroft, Ullman, “Introduction to Automata Theory, Language and Computation”, NerosaPublishing House 2. Cohen D. I. A., “Introduction to Computer theory”, John Wiley & Sons	

SET/CS/BT/C405	Data Communication and Computer Networks	
Course Objective	1. Build an understanding of the fundamental concepts of computer networking. 2. Familiarize the student with the basic taxonomy and terminology of the computer networking area. 3. Introduce the student to advanced networking concepts, preparing the student for entry Advanced courses in computer networking 4. Independently understand basic computer network technology. 5. Identify the different types of network topologies and protocols. 6. Enumerate the layers of the OSI model and TCP/IP. 7. Explain the function(s) of each layer.	
Course Outcome	1. Understand the concepts of Data Communication. 2. Study the functions of OSI Layers. 3. Familiarize with the Transmission Media, Flow Control and Error Detection & Correction. 4. Understand fundamental concepts in Routing, Addressing & working of Transport Protocols. 5. Gain familiarity with common networking & Application Protocols. 6. Understand Wireless LANs & Wireless Sensor Networks Operation.	
Module Name	**Content**	**No. of Hrs.**
Introduction	Introduction to Computer Networking: Use, advantage, structure of the communications network topologies the telephone network, analog to digital communication. Network classes, Repeaters Hub, Bridges, Switches, Routers, Gateways B-routers.	6
Data Communications	Fundamentals: Layered Network Architecture, Communication Between Analog Computers & Terminals Layered Protocols, Network & The OSI Models, Traffic control and accountability wide area and local area networks, connection oriented and connectionless networks, classification of communication protocols polling/selection systems, design problems, communication between layers, ISO standard. Transmission Media: Guided, Unguided; Transmission Impairments and Channel Capacity; Transmission of Digital Data, Interfaces-DTE-DCE, MODEM, The telephone network system and DSL technology;	8
Data link layer:	Introduction, Framing, and Error – Detection and Correction – Parity – LRC – CRC Hamming code, Flow and Error Control, Noiseless Channels, Noisy Channels, HDLC, Medium Access sub layer: ALOHA, CSMA/CD, IEEE LAN Standards Random access, Controlled access, Channelization. Data Link Protocols: Synchronous, Asynchronous Protocols, Point-to-Point Protocol(PPP) Switching Communication Networks: Circuit switching; Packet switching; Routing in packet switched networks; X.25; Frame Relay, ATM, ISDN.	10
Network Layer	Network Layer Design Issues, Routing Algorithms, Network Layer Protocols IP Addressing, CIDR & NAT, IP layer protocols (ICMP, ARP, RARP, DHCP, and BOOTP) and IPv6, TCP/IP and internetworking, Network Devices.	10
Transport layer and Application layer	Process to Process Delivery, UDP and TCP protocols, Data Traffic, Congestion Control, QoS, Integrated Services, Differentiated Services. Domain name space, DNS in internet, electronic mail, SMTP, FTP, WWW, HTTP, SNMP.	8
Total No. of Hours		**42**
Text Books	1. Data communication & Networking by Bahrouz Forouzan. 2. Stallings, W. (2010), Data and Computer Communications, Pearson.	
References	1. J. Kurose, K. Ross, Computer Networking: A Top - Down Approach, Pearson 2. Tannanbaum, A.S.: Computer Network, PHI 3. Black : Computer Network; Protocols, Standards and Interface PHI	

SET/CS/BT/C406 OBJECT ORIENTED PROGRAMMING USING C++ LAB		
Course Objective	1. **T**o introduce students to the fundamental programming constructs and techniques. 2. To provide hands-on experience in implementing programs using these constructs and techniques. 3. To develop the ability to design and implement programs that solve practical problems. 4. To develop the ability to use programming concepts in the context of object-oriented programming (OOP). 5. To provide students with a foundation for further study in computer science and related fields.	
Course Outcome	1. Students will demonstrate proficiency in using input and output statements effectively. 2. Students will be able to implement control statements to manage the flow of programs. 3. Students will develop the ability to define and use functions to modularize code. 4. Students will demonstrate competency in implementing arrays to store and manipulate data. 5. Students will be capable of implementing classes along with constructors and destructors for object-oriented programming. 6. Students will gain experience in performing file operations such as reading from and writing to files. 7. Students will understand and apply object-oriented programming concepts like inheritance, polymorphism, encapsulation, friend functions, and static functions effectively in their programs.	
Module Name	**Content**	**No. of Hrs.**
Experiments / Spice Simulations	1. Implementation of input and output statements. 2. Implementation of control statements. 3. Implementation of functions. 4. Implementation of array 5. Implementation of Classes and Constructor and Destructor. 6. Implementation of files. 7. Implementation of OOP‟s Concepts (Inheritance, Polymorphism, Encapsulation, Friend and Static Functions)	**3x12**
	Total No. of Hours	**36**

SET/CS/BT/C407 OPERATING SYSTEMS LAB		
Course Objective	1. To introduce students to the Bourne Shell commands and constructs used in UNIX environments. 2. To provide hands-on experience in using the Bourne Shell commands and constructs for file management and scripting. 3. To develop the ability to write moderately complex Shell scripts to automate tasks and solve practical problems. 4. To teach tracing mechanisms for debugging and exporting variables for use in other scripts. 5. To provide students with the ability to execute programs written in C under the UNIX environment. 6. To develop an understanding of how to customize the user environment using the ".profile" script. 7. To provide a foundation for further study in UNIX/Linux system administration and scripting.	
Course Outcome	8. Students will demonstrate proficiency in using basic Bourne Shell commands such as cat, grep, ls, ps, chmod, finger, etc. 9. Students will be able to utilize Bourne Shell constructs like if-then, if-then-else, for, while, until, and case for program control. 10. Students will gain an understanding of tracing mechanisms, shell variables, command substitution, and other advanced shell scripting concepts. 11. Students will demonstrate the ability to perform file operations and test conditions on numeric values, file types, and character strings within Shell scripts. 12. Students will develop the skill to write moderately complex Shell scripts to automate tasks. 13. Students will learn how to make Shell scripts executable using appropriate permissions. 14. Students will create a customized ".profile" script to tailor their user environment according to their preferences. 15. Students will be able to compile and execute C programs in a UNIX environment, gaining familiarity with the compilation and execution process.	
Module Name	**Content**	**No. of Hrs.**
Module 1	1. Demonstrate how to use the following Bourne Shell commands: cat, grep, ls ,more, ps, chmod, finger etc. 2. Use the following Bourne Shell constructs: test, if then, if then else, if then el if, for, while, until, and case. 3. Learn tracing mechanisms (for debugging), user variables, Bourne Shell variables, read-only variables, positional parameters, reading input to a Bourne Shell script, command substitution, comments, and exporting variables. 4. In addition, test on numeric values, test on file type, and test on character strings are covered. Copy, move, and delete files and directories. 5. Write moderately complex Shell scripts. 6. Make a Shell script executable 7. Create a ".profile" script to customize the user environment.. 8. Execute programs written in C under UNIX environment	3x12
	Total No. of Hours	**36**

SET/CS/SC/C408	MINI PROJECT	
Course Objective	1. Upon completing Mini Project-1, students will have gained the ability to apply the programming concepts and techniques learned in C/C++ to solve real life problem.	
Course Outcome	1. To provide students with an opportunity to apply the programming concepts and techniques learned in C/C++ to solve a practical problem. 2. To develop the ability to design and implement a program using C/C++ to solve a real-world problem. 3. To develop the ability to use software development tools and techniques, such as version control, debugging, and testing, in the context of a larger programming project. 4. To provide students with the experience of working in a team to develop a program. 5. To develop communication and presentation skills through the documentation and presentation of the Mini Project-1. 6. To provide students with the opportunity to apply critical thinking and problemsolving skills to a real-world problem. 7. To provide a foundation for further study and work in software development and programming.	
Module Name	**Content**	**No. of Hrs.**
Module 1	Mini Project-1 shall be based on C/C++.	3x12
	Total No. of Hours	**36**

Semester V

S. No.	Category	Course Code	Course Title	L	T	P	Contact Hrs./Week	Credits
1	**Core Subjects**	SET/CS/BT/C501	Database Management System	3	1	-	4	4
2		SET/CS/BT/C502	Design and Analysis of Algorithms	3	1	-	4	4
3		SET/CS/BT/C503	Software Engineering	3	1	-	4	4
4			Program Elective-1	3	1	-	4	4
5	**Open Elective/Inter-disciplinary Subject**		Open Elective-1	3	1		4	4
6	**Core Subjects Based Labs**	SET/CS/BT/C504	Database Management System Lab	-		1	3	1
7		SET/CS/BT/C505	Design and Analysis of Algorithms Lab			1	3	1
8	**Extracurricular Courses/Compulsory course**	SET/CS/BT/M506	*Culture, traditions and moral values/ Yoga Practices	-	-	1	4	2
9	**Skill Course**	SET/CS/SC/C507	Python Lab			1	4	2
Total				15	5	4	34	26

* University will prepare a course with focus on Indian/ Regional culture studies.

Elective and Open Elective Courses

S. No.	Category	Course Code	Course Title	L	T	P	Contact Hrs./Week	Credits
1	**Program Elective-1**	SET/CS/BT/E501	Distributed Computing	3	1	-	4	4
2		SET/CS/BT/E502	Graph Theory	3	1	-	4	4
3		SET/CS/BT/E503	Principles of Programming Languages	3	1	-	4	4
1	**Open Elective-1**	SET/CS/BT/OE501	Java Programming	2	1	1	4	4
2		SET/CS/BT/OE502	Project Management	3	1		4	4
3		SET/CS/BT/OE503	Optimization Techniques	3	1		4	4

Semester V

DATABASE MANAGEMENT SYSTEM (SET/CS/BT/C501)		
Course Objective	1. Understand the fundamental differences between database systems and file systems, and grasp the key concepts of database system architecture. 2. Master the Entity-Relationship Model, including concepts such as super keys, candidate keys, and primary keys, and learn to reduce ER diagrams to relational tables. 3. Develop proficiency in database design and normalization techniques, including functional dependencies and normalization up to BCNF. **4.** Gain knowledge of transaction processing concepts, including serializability, recoverability, concurrency control techniques like locking and time stamping protocols, and an introduction to SQL for data manipulation and querying.	
Course Outcome	1. Understand the difference between database systems and file systems, and grasp key concepts like data models, schema, and data independence. 2. Master entity-relationship modeling techniques, including super keys, primary keys, and the reduction of ER diagrams to relational tables. 3. Gain proficiency in database design and normalization, including functional dependencies and normalization forms up to BCNF. **4.** Develop skills in transaction processing, concurrency control, and SQL, enabling effective database management and application development.	
Module Name	**Content**	**No. of Hrs.**
Fundamental Concepts	Database system Vs file system, Database system concepts and architecture, Data models schema and instances, data independence and data base languageand interfaces, Data definitions language, DML, Overall Database Structure.	8
Entity Relationship Model	ER model concepts, Concepts of Super Key, candidate key, primary key, Generalization, aggregation, reduction of an ER diagrams to tables, extended ER model, relationships of higher degree. Relational data Model and Language, integrity constraints, relational algebra, relational calculus, tuple And domain calculus.	6
Data Base Design & Normalization	Functional dependencies, normal forms, first, second, third normal forms, BCNF, inclusion dependences, loss less join decompositions, normalization using FD, MVD, and JDs, alternative approaches to database design. Transaction.	10
Processing Concepts	Transaction system, Testing of serializability, Serializability of schedules, conflict & view serializable schedule, recoverability, Recovery from transaction failures, log based recovery, checkpoints, deadlock handling. Concurrency control, locking Techniques for concurrency control, Time stamping protocols for concurrency control, validation based protocol, Multiple.	12
Introduction to SQL	Characteristics of SQL. Advantage of SQL. SQL data types and literals. Types of SQL commands. SQL operators and their procedure. Tables, views and indexes. Queries and sub queries. Aggregate functions. Insert, update and delete operations. Joins, Unions, Intersection, Minus, Cursors in SQL Granularity, Multi version schemes, Recovery with concurrent transaction.	6
	Total No. of Hours	**42**

Text Books	1. Korth, Silbertz, Sudarshan, “Database Concepts”, McGraw Hill
References	2. Elmasri, Navathe, “Fundamentals Of Database Systems”, Addision Wesley 3. Date C.J. "An Introduction to Database System". Addision Wesley

DESIGN AND ANALYSIS OF ALGORITHMS (SET/CS/BT/C502)		
Course Objective	1. Provide a comprehensive understanding of algorithm design and analysis, including the analysis of algorithmic efficiency and the application of the Master's Theorem. 2. Explore various sorting algorithms such as Heap sort, Quick sort, and linear time sorting methods, along with order statistics. 3. Delve into advanced data structures like Red-Black Trees, B-Trees, and heaps, including their augmentation for specific purposes. 4. Cover dynamic programming, greedy algorithms, and backtracking techniques for problem-solving, with an emphasis on algorithmic design paradigms. 5. Study graph algorithms including minimum spanning trees, shortest paths, maximum flow, and selected advanced topics like randomized algorithms and NP completeness.	
Course Outcome	1. Demonstrate proficiency in analyzing algorithms and determining their efficiency through growth functions and the Master's Theorem. 2. Implement various sorting algorithms and understand their performance characteristics. 3. Design and implement advanced data structures such as Red-Black Trees and B-Trees for efficient data management. 4. Apply dynamic programming, greedy algorithms, and backtracking techniques to solve a variety of computational problems. 5. Develop a strong understanding of graph algorithms and their applications in solving real-world problems, including optimization and network flow.	
Module Name	**Content**	**No. of Hrs.**
Fundamental Concepts	Algorithms, analysis of algorithms, Growth of Functions, Master's Theorem, Designing of Algorithms.	10
Sorting and order Statistics	Heap sort, Quick sort, Sorting in Linear time, Medians and Order Statistics. Advanced Data Structure: Red-Black Trees, Augmenting Data Structure. B-Trees, Binomial Heaps, Fibonacci Heaps, Data Stricture for Disjoint Sets.	10
Design and Analysis	Dynamic Programming, Greedy Algorithms, Amortized Analysis, Back Tracking.	10
Graph Algorithms	Elementary Graphs Algorithms, Minimum Spanning Trees, Single source Shortest Paths, All-Pairs Shortest Paths, Maximum Flow, and Traveling Salesman Problem. Selected Topics: Randomized Algorithms, String Matching, NP Completeness, Approximation Algorithms.	12
	Total No. of Hours	**42**
Textbooks	1. Coreman, Rivest, Lisserson, “Algorithm", PHI.	
References	2. Basse, "Computer Algorithms: Introduction to Design & Analysis", Addision Wesley. 3. Horowitz & Sahani, "Fundamental of Computer Algorithm", Galgotia.	

SOFTWARE ENGINEERING (SET/CS/BT/C503)		
Course Objective	1. Provide an understanding of software engineering processes, including software components, characteristics, and the historical context of the software crisis. 2. Explore various software development models such as Waterfall, Spiral, and Iterative Enhancement models, emphasizing their similarities and differences from conventional engineering processes. 3. Introduce software quality attributes, quality assurance techniques, verification and validation processes, and frameworks for ensuring software quality. 4. Cover the process of software requirement specifications and design, including elicitation, analysis, documentation, and architectural design using various modeling techniques. **5.** Discuss software measurement and metrics, including size-oriented measures, estimation techniques, software testing strategies, and the importance of software maintenance and configuration management.	
Course Outcome	1. Demonstrate knowledge of software engineering processes and their application in software development projects. 2. Analyze and evaluate different software development models to determine their suitability for various project requirements. 3. Apply software quality assurance techniques to ensure the reliability, maintainability, and efficiency of software products. 4. Design software systems using appropriate modeling techniques and design strategies, considering factors like modularity, coupling, and cohesion. **5.** Utilize software measurement and metrics to estimate project parameters accurately, conduct effective software testing, and manage software maintenance activities efficiently.	
Module Name	**Content**	**No. of Hrs.**
Introduction	Software Components, Software Characteristics, Software Crisis, Software Engineering Processes, Similarity and Differences from Conventional Engineering Processes, Software Quality Attributes. Software Quality Assurance, Verification and Validation, SQA Plans, Software Quality Frameworks. Software Development Models, Water Fall Model, Prototype Model, Spiral Model, Evolutionary Development Models, Iterative Enhancement Models.	10
Software Requirement Specifications and Design	Elicitation, Analysis, Documentation, Review and Management of User Needs, Feasibility Study, Information Modeling, Data Flow Diagrams, Entity Relationship Diagrams, Decision Tables, SRS Document. Architectural Design, Low Level Design: Modularization, Design Structure Charts, Pseudo Codes, Flow Charts, Coupling and Cohesion Measures. Design Strategies: Function Oriented Design, Object Oriented Design, Top-Down and Bottom-Up Design.	10
Software Measurement and Metrics	Various Size Oriented Measures, Halestead's Software Science, Function Point (FP) Based Measures, Cyclomatic Complexity Measures, Control Flow Graphs. Estimation of Various Parameters such as Cost, Efforts, Schedule/Duration, Constructive Cost Models (COCOMO).	8
Software Testing	Testing Objectives, Unit Testing, Integration Testing, Acceptance Testing, Regression Testing, Testing for Functionality and Testing for Performance, Top-Down and Bottom-Up Testing Strategies, Structural Testing, Functional Testing, Test Data Suit Preparation, Alpha and Beta Testing of Products. Static Testing Strategies.	8
Software Maintenance	Need for Maintenance, Categories of Maintenance: Preventive, Corrective and Perfective Maintenance, Cost of Maintenance, Software Re-Engineering, Reverse Engineering. Software Configuration Management Activities. Change Control Process, Software Version Control, An Overview of CASE Tools.	6
	Total No. of Hours	**42**
Textbooks	1. K. K. Aggarwal and Yogesh Singh, Software Engineering, New Age International Publishers.	
References	**1.** R. S. Pressman, Software Engineering: A Practitioners Approach, McGraw Hill. **2.** Ian Sommerville, Software Engineering, Addison Wesley.	

Database Management System Lab (SET/CS/BT/C504)		
Lab Objective	1. Familiarize students with Data Definition Language (DDL) and Data Manipulation Language (DML) in SQL. 2. Enable students to write SQL queries using logical operators, SQL operators, and functions. 3. Provide hands-on experience in relational algebra operations and querying data from multiple tables. 4. Introduce students to sub queries, nested queries, and advanced SQL concepts like PL/SQL programming. 5. Equip students with practical skills in creating views, cursors, triggers, and implementing database integrity constraints.	
Lab Outcome	1. Proficiency in writing SQL queries for defining and manipulating database structures. 2. Ability to construct SQL queries using logical and SQL operators for data retrieval and manipulation. 3. Competence in performing relational algebra operations and querying data from multiple tables using various types of joins. 4. Skill in writing complex SQL queries involving sub queries and nested queries to extract desired information from databases. 5. Capability to create and manage database objects like views, cursors, triggers, and implement business rules using PL/SQL programming in Oracle databases.	
Module Name	**Content**	**No. of Hrs.**
Module 1	1. Write the queries for Data Definition and Data Manipulation language. 2. Write SQL queries using Logical operators (=, <,>, etc.). 3. Write SQL queries using SQL operators (Between…. AND, IN (List), Like,ISNULL and also with negating expressions). 4. Write SQL query using character, number, date and group functions. 5. Write SQL queries for Relational Algebra (UNION, INTERSECT, and MINUS, etc.). 6. Write SQL queries for extracting data from more than one table (Equip-Join,Non-Equip-Join , Outer Join) 7. Write SQL queries for sub queries, nested queries. 8. Write programs by the use of PL/SQL. 9. Concepts for ROLL BACK, COMMIT & CHECK POINTS. 10. Create VIEWS, CURSORS, and TRIGGRS & write ASSERTIONS, Create FORMS and REPORTS.	3x12
	Total No. of Hours	**36**

DESIGN AND ANALYSIS OF ALGORITHMS LAB (SET/CS/BT/C505)		
Lab Objective	1. Introduce students to fundamental algorithmic design paradigms including Divide and Conquer, Greedy Method, Dynamic Programming, and Backtracking. 2. Provide hands-on experience in implementing classic algorithms and solving real-world problems using these paradigms. 3. Familiarize students with common sorting and searching algorithms along with selection algorithms. 4. Develop students' proficiency in analyzing algorithmic complexities and performance. 5. Encourage critical thinking and problem-solving skills through practical algorithm implementation and analysis.	
Lab Outcome	1. Mastery in implementing Divide and Conquer algorithms such as Quick Sort, Merge Sort, and Stassen's Matrix Multiplication. 2. Competence in solving optimization problems using the Greedy Method, including the Knapsack Problem and Minimal Spanning Trees. 3. Proficiency in Dynamic Programming techniques to solve complex problems like the 0/1 Knapsack Problem and Traveling Salesperson Problem. 4. Ability to apply Backtracking algorithms to solve problems like the N-Queens Problem and Graph Coloring Problem. 5. Understanding and practical application of common sorting and searching algorithms like Insertion Sort, Heap Sort, Binary Search, and Sequential Search, along with selection algorithms for finding minimum/maximum elements and the Kth smallest element.	
Module Name	**Content**	**No. of Hrs.**
Module 1	1. Divide and conquer method (quick sort, merge sort, Strassen's matrix multiplication), 2. Greedy method (knapsack problem, job sequencing, optimal merge patterns, minimal spanning trees). 3. Dynamic programming (multistage graphs, OBST, 0/1 knapsack, traveling salesperson problem). 4. Back tracking (n-queens problem, graph coloring problem, Hamiltonian cycles). 5. Sorting : Insertion sort, Heap sort, Bubble sort 6. Searching : Sequential and Binary Search 7. Selection : Minimum/ Maximum, Kth smallest element	3x12
	Total No. of Hours	**36**

PYTHON LAB (SET/CS/SC/C506)		
Lab Objective	1. Introduce students to Python programming language and its basic data types. 2. Provide hands-on experience in solving problems using Python. 3. Familiarize students with common programming tasks such as finding the union of lists, counting occurrences of words in a string, and finding the largest number among three numbers. 4. Develop students' proficiency in writing functions and solving problems using functions. 5. Encourage students to apply built-in functions like map () to manipulate data in Python.	
Lab Outcome	1. Understanding of Python syntax and basic data types including lists, strings, and numbers. 2. Ability to write Python programs to solve various programming tasks such as finding the union of lists and counting word occurrences. 3. Proficiency in solving simple arithmetic problems and performing comparisons in Python. 4. Competence in writing and utilizing functions to solve problems, such as finding factorial or checking for equality between two lists. 5. Capability to utilize built-in functions like map () effectively to perform operations on iterable objects in Python.	
Module Name	**Content**	**No. of Hrs.**
Module 1	1. Introduction to python programming and python datatypes. 2. Write a python program to find the union of two lists. 3. Write a python program to count the occurrences of each word in a given string sentence. 4. Write a python program to find largest number among three numbers. 5. Write a python program to check whether the given string is palindrome or not. 6. Write a python program to find factorial of a given number using functions. 7. Write a Python function that takes two lists and returns true if they are equal otherwise false. 8. Write a program for map () function to double all the items in the list.	3x12
	Total No. of Hours	**36**

Program Elective Courses

<table>
<tr><td colspan="3">DISTRIBUTED SYSTEMS (SET/CS/BT/E501)</td></tr>
<tr><td>Course Objective</td><td colspan="2">1. Provide a comprehensive understanding of architectural models and fundamental concepts in distributed systems, including theoretical foundations and limitations.
2. Explore advanced topics such as distributed mutual exclusion, deadlock prevention, and agreement protocols, along with their applications.
3. Cover distributed file systems, resource sharing challenges, and communication between distributed objects, including remote procedure calls.
4. Discuss distributed transactions, including concurrency control methods, atomic commit protocols, and techniques for handling distributed deadlocks and transaction recovery.
5. Introduce distributed algorithms for communication protocols, routing, packet switching, and election algorithms.</td></tr>
<tr><td>Course Outcome</td><td colspan="2">1. Demonstrate a deep understanding of architectural models and theoretical foundations underlying distributed systems.
2. Analyze and evaluate different approaches to distributed mutual exclusion and deadlock prevention, along with their performance metrics.
3. Design and implement distributed file systems and communication mechanisms between distributed objects, including remote invocation.
4. Apply various methods for concurrency control and transaction management in distributed environments.
5. Develop proficiency in designing and implementing distributed algorithms for communication, routing, and fault-tolerant services, considering factors like scalability and reliability.</td></tr>
<tr><td>Module Name</td><td>Content</td><td>No. ofHrs.</td></tr>
<tr><td>System Models</td><td>Architectural Models, Fundamental Models, Theoretical Foundation for Distributed System, Limitation ofDistributed system, Absence of global clock, Shared memory, Logical clocks, Lamport’s & vectors logical clocks, Causal ordering of messages, Global state, Termination detection. Resource sharing and the Web Challenges. Distributed Objects and Remote Invocation, Communication between distributed objects, Remote procedure call. Distributed File Systems, architecture, Sun Network File System, The Andrew File System.</td><td>10</td></tr>
<tr><td>Distributed Mutual Exclusion and Deadlock</td><td>Classification of distributed mutual exclusion, Requirement of mutual exclusion theorem, Token based and non-token based algorithms, Performance metric for distributed mutual exclusion algorithms.
Resource vs. Communication deadlocks, Deadlock prevention, Avoidance, detection & resolution, Centralized dead lock detection, Distributed dead lock detection, Path pushing algorithms, edge chasing algorithms.</td><td>8</td></tr>
<tr><td>Agreement Protocols</td><td>Classification of Agreement Problem, Byzantine agreement problem, Consensus problem, Interactive Consistency Problem, Solution to Byzantine Agreement problem, Application of Agreement problem.</td><td>8</td></tr>
<tr><td>Distributed Transactions</td><td>Nested transactions, Locks, Optimistic Concurrency control, Timestamp ordering, Comparison of methods for concurrency control, Flat and nested distributed transactions, Atomic Commit protocols, Concurrency control in distributed transactions, Distributed deadlocks, Transaction recovery. Replication, Fault - tolerant Services, highly available services, Transactions with replicated data.</td><td>10</td></tr>
<tr><td>Distributed Algorithms</td><td>Communication protocols, Balanced sliding window protocol, Routing algorithms, Destination based Routing, Deadlock free Packet switching, Wave & traversal algorithms, Election algorithm.</td><td>6</td></tr>
<tr><td colspan="2">Total No. of Hours</td><td>42</td></tr>
<tr><td>Text Books</td><td colspan="2">1. Singhal & Shivaratri, "Advanced Concept in Operating Systems", McGraw Hill.</td></tr>
<tr><td>References</td><td colspan="2">1. Coulouris, Dollimore, Kindberg, "Distributed System: Concepts and Design”, Pearson Ed.
2. Gerald Tel, "Distributed Algorithms", Cambridge University Press.</td></tr>
</table>

	GRAPH THEORY (SET/CS/BT/E502**)**	
Course Objective	1. Provide a comprehensive understanding of graph theory concepts including sub graphs, walks, paths, circuits, and various operations on graphs. 2. Explore properties of trees and fundamental circuits, including spanning trees, Hamiltonian paths, and the Traveling Salesman problem. 3. Cover cuts-sets, cut vertices, network flows, and planar graphs along with their duals and detection of planarity. 4. Introduce vector space representations of graphs, matrix representations, and their relationships, including incidence matrices and adjacency matrices. 5. Discuss coloring, covering, partitioning, matching, and enumeration techniques in graph theory, including the application of Polya's Counting Theorem.	
Course Outcome	1. Demonstrate proficiency in analyzing basic graph properties and operations, such as sub graphs, walks, and connectivity. 2. Apply graph theory concepts to solve problems related to trees, spanning trees, and Hamiltonian paths. 3. Understand cuts-sets, network flows, planar graphs, and their duals, and apply them to solve related problems. 4. Utilize vector space and matrix representations to analyze graphs and their properties effectively. 5. Apply coloring, covering, and partitioning techniques to solve problems related to graph theory, and enumerate graphs using Polya's Counting Theorem.	
Module Name	**Content**	**No. of Hrs.**
Module 1	Graphs: Sub Graphs, Basic Properties, Example of Graphs & Their Sub Graphs, Walks, Path & Circuits, Connected Graphs, Disconnected Graphs and Components, Euler Graphs, Various Operation on Graphs, Hamiltonian Paths and Circuits, The Traveling Salesman problem, Trees and Fundamental Circuits: Basic Properties of Tree, Distance and Centers, Radius and Diameters, Pendent Vertices, Rooted and Binary Trees, On Counting Trees, Spanning Trees, Fundamental Circuits, Finding Spanning Trees of a Graph and a Weighted Graph: Prim's Algorithm,Kruskal's Algorithm.	12
Module 2	Cuts-sets and Cut Vertices: Cut-sets , Properties of Cut-sets, Fundamental Circuits and Cut-sets,Connectivity and Separability, Network Flows, Planer and Dual Graphs: Combinatorial and Geometric Dual, Kuratowski to graphs detection of Planarity, geometric dual, some more criterion of planarity, thickness and crossings.	10
Module 3	Vector Space of a Graph and Vectors: Basis Vector, Cut-set Vector, Circuit Vector, Circuit and Cut-set verses Subspaces, Orthogonal Vectors and Subspaces, Matrix Representation of Graphs: Incidence Matrix of Graph, Sub Matrices of A(G), Circuit Matrix,Cut-set Matrix, Path Matrix and Relationships among Af, Bf, and Cf, Fundamental Circuit Matrix and Rank of B, Adjacency Matrices, Rank-Nullity Theorem	8
Module 4	Coloring, Covering and Partitioning: Chromatic Number, Chromatic Partitioning, Chromatic Polynomials, Matching, Covering, Four Color Problem, Directed Graphs, Some Type of Directed Graphs, Directed Paths, and Connectedness, Euler Digraphs, Trees with Directed Edges, Fundamental Circuits in Digraph, Matrices A, B and C of Digraphs Adjacency Matrix of a Digraph,	8
Module 5	Enumeration: Types of Enumeration, Counting of Labeled and Unlabeled Trees, Polya's CountingTheorem, Graph Enumeration with Polya's Theorem. Graph Theoretic Algorithms	6
	Total No. of Hours	**44**
Textbooks	1. Deo, N, "Graph Theory", PHI	
References	1. Harary, F, "Graph Theory", Narosa 2. Bondy and Murthy, "Graph Theory and Application", Addison Wesley.	

PRINCIPLES OF PROGRAMMING LANGUAGES (SET/CS/BT/E503)		
Course Objective	1. Provide an understanding of the characteristics and factors influencing the evolution of programming languages, along with developments in programming methodologies. 2. Introduce the structure and operations of programming language processors, including translators, and the concept of virtual computers. 3. Cover data types in programming languages, including elementary data types, structured data types, and abstraction mechanisms like abstract data types. 4. Explore sequence control mechanisms in programming languages, including implicit and explicit sequence control, recursion, exception handling, and concurrent execution. 5. Discuss storage management in programming languages, including static and dynamic memory allocation, stack-based and heap-based storage management.	
Course Outcome	1. Demonstrate knowledge of the characteristics and factors influencing programming language evolution, enabling informed decision-making in language selection and design. 2. Understand the structure and operation of programming language processors, aiding in the development of efficient translation tools and virtual machines. 3. Proficiency in working with data types in programming languages, including the declaration, assignment, and manipulation of variables, constants, and structured data types. 4. Ability to implement various sequence control mechanisms, such as recursion, exception handling, and concurrency, to develop robust and efficient programs. 5. Competence in managing memory resources efficiently through static and dynamic storage management techniques, facilitating the development of scalable and reliable software systems.	
Module Name	**Content**	**No. of Hrs.**
Introduction	Characteristics of programming Languages, Factors influencing the evolution of Programming language, developments in programming methodologies, desirable featuresand design issues.	6
Programming Language Processors	Structure and operations of translators, software simulated computer, syntax, semantics, Structure, virtual computers, binding and binding time.	8
Data Types	Data object variables, constants, data types, elementary data types, declaration, assignment and initialization, enumeration, characters, strings. Structured data type and objects: Specification of data structured types, vectors and arrays, records, variable size data structure, pointers and programmer constructed data structure, Sets files. Abstractions, encapsulations, information hiding, sub programmes, abstract data types	12
Sequence Control	Implicit and Explicit sequence control, sequence control with within expression and statements, recursive sub programmes, exception handling, coroutines, Scheduled sub programmes, concurrent execution. Data control referencing environments, static and dynamic scope, local data local data referencing environment,	8
Storage Management	Major run time requirements, storage management phases, static storage management, stackbased, heap based storage management.	8
	Total No. of Hours	**42**
Textbooks	1. Terrance W Pratt, "Programming Languages: Design and Implementation" PHI 2. E Horowitz, "Programming Languages", 2nd Edition, Addison Wesley	
References	1. Sebesta, "Concept of Programming Language", Addison Wesley 2. Fundamentals of Programming Languages, Galgotia.	

Open Elective Courses

Java Programming (SET/CS/BT/OE501**)**		
Course Objective	1. Provide an understanding of the fundamental features of Java programming language, including its object-oriented nature and platform independence. 2. Introduce the Java Virtual Machine (JVM) and byte code interpretation, along with key concepts such as data types, variables, arrays, expressions, operators, and control structures. 3. Cover advanced topics in Java such as objects, classes, abstract classes, static classes, inner classes, packages, wrapper classes, interfaces, and access control. 4. Explore exception handling mechanisms in Java, including exception hierarchy, try-catch-finally blocks, and I/O operations using Java IO package. 5. Introduce database connectivity in Java using JDBC, along with concepts like Java Beans, Servlets, Java Server Pages (JSP), and their applications in web development.	
Course Outcome	1. Demonstrate proficiency in using Java's object-oriented features to develop robust and scalable software applications. 2. Understand the architecture of Java Virtual Machine (JVM) and its role in executing Java byte code. 3. Implement various control structures and data types in Java to manipulate data and control program flow effectively. 4. Develop skills in handling exceptions and performing input/output operations using Java IO package, contributing to error-free and efficient Java programs. 5. Gain practical experience in database connectivity using JDBC, along with web development concepts like Servlets, JSP, and Java Beans, enabling the development of dynamic and interactive web applications.	
Module Name	**Content**	**No. of Hrs.**
Java Fundamentals	Features of Java, OOPs concepts, Java virtual machine, Reflection byte codes, Bytecode interpretation, Data types, variable, arrays, expressions, operators, and controlstructures, Objects and classes. Abstract classes, Static classes, Inner classes, Packages, Wrapper classes, Interfaces, This ,Super, Access control	10
Exception handling	Exception as objects ,Exception hierarchy, Try, catch, finally, Throw, throws, IO package, Input streams, Output streams, Object serialization, De-serialization, Sample programs on IO files, Filter and pipe streams, Multi-threading, Thread Life cycle, Multi-threading advantages and issues, Simple thread program, Thread synchronization, GUI, Introduction to AWT programming, Layout and component managers, Event handling, Applet class, Applet life-cycle, Passing parametersembedding in HTML, Swing components – J Applet, J Button, J Frame.	12
Java Beans and Web Servers	Introduction to Java Beans, Advantage, Properties, BDK, Introduction to EJB, Java Beans API Introduction to Servlets, Lifecycle, JSDK, Servlet API, Servlet Packages: HTTP package, Working with Http request and response, Security Issues. JSP: Introduction to JSP, JSP processing, JSP Application Design, Tomcat Server, Implicit JSP objects, Conditional Processing, Declaring variables and methods, Error Handling and Debugging, Sharing data between JSP pages- Sharing Session and Application Data.	10
Database Connectivity	Database Programming using JDBC, Studying Javax.sql. package, accessing a database from a JSP page, Application-specific Database Action, Developing Java Beansin a JSP page, introduction to Struts framework	10
Java Fundamentals	Features of Java, OOPs concepts, Java virtual machine, Reflection byte codes, Bytecode interpretation, Data types, variable, arrays, expressions, operators, and controlstructures, Objects and classes. Abstract classes, Static classes, Inner classes, Packages, Wrapper classes, Interfaces, This ,Super, Access control	10
	Total No. of Hours	**42**

Textbooks	1. Java – Balaguruswamy
References	2. Java Programming John P. Flynt Thomson 2nd 3. Java Programming Language Ken Arnold Pearson 4. The complete reference JAVA2, Herbert schildt. TMH 5. Big Java, Cay Horstmann 2nd edition, Wiley India Edition

PROJECT MANAGEMENT (SET/CS/BT/OE502)		
Course Objective	1. Provide a comprehensive understanding of software project management fundamentals, including the project management cycle and its objectives. 2. Introduce techniques and methods for software project planning, estimation, and decision-making, along with the structure of a software project management plan. 3. Explore project organization strategies, including work breakdown structures (WBS), project scheduling, and various scheduling techniques. 4. Cover the dimensions of project monitoring and control, including earned value analysis and software reviews, to ensure project success. 5. Discuss software quality assurance and testing principles, including test plans, test cases, and quality metrics, along with software configuration management techniques and risk management.	
Course Outcome	1. Demonstrate proficiency in software project management fundamentals, enabling effective planning, execution, and control of software projects. 2. Develop skills in software project estimation and decision-making, contributing to accurate project planning and resource allocation. 3. Apply project organization strategies such as work breakdown structures and scheduling techniques to effectively manage project resources and timelines. 4. Utilize earned value analysis and other project monitoring techniques to track project progress and ensure adherence to budget and schedule. 5. Implement software quality assurance and testing principles to ensure the delivery of high-quality software products, along with effective software configuration management practices to manage changes and mitigate risks throughout the project lifecycle.	
Module Name	**Content**	**No. of Hrs.**
Software Project Planning	Fundamentals of Software Project Management (SPM), Need Identification, Vision and Scope document,Project Management Cycle, SPM Objectives, Management Spectrum, SPM Framework, Software Project Planning, Planning Objectives, Project Plan, Types of project plan, Structure of a Software ProjectManagement Plan, Software project estimation, Estimation methods, Estimation models, Decision process.	10
Project Organization	Project Elements, Work Breakdown Structure (WBS), Types of WBS, Functions, Activities and Tasks, Project Life Cycle and Product Life Cycle, Ways to Organize Personnel, Project schedule, Scheduling Objectives, Building the project schedule, Scheduling terminology and techniques. Network Diagrams, PERT, CPM, Bar Charts: Milestone Charts, Gantt Charts.	8
Project Monitoringand Control	Dimensions of Project Monitoring & Control, Earned Value Analysis, Earned Value Indicators: BudgetedCost for Work Scheduled, Cost Variance, Schedule Variance, Cost Performance Index, Schedule Performance Index, Interpretation of Earned Value Indicators, Error Tracking, Software Reviews.	8
Software Quality Assurance and Testing	Testing Objectives, Testing Principles, Test Plans, Test Cases, Types of Testing, Levels of Testing, Test Strategies, Program Correctness, Program Verification & validation, Testing Automation & Testing Tools,Concept of Software Quality, Software Quality Attributes, Software Quality Metrics and Indicators, The SEI Capability Maturity Model CMM), SQA Activities, Statistical quality assurance, Clean room process.	8
Software Configuration Management	Software Configuration Items and tasks, Baselines, Plan for Change, Change Control, Change Requests Management, Version Control, Risk Management: Risks and risk types, Risk Breakdown Structure (RBS),Risk Management, Risk identification, Risk analysis, Risk planning, Risk monitoring, Cost Benefit Analysis.	8
	Total No. of Hours	**42**
Text books	1. M. Cotterell, “Software Project Management”, TMH	
References	1. S. A. Kelkar, “Software Project Management”, PHI 2. Royce, “Software Project Management”, Pearson Education 3. Kieron Conway, “Software Project Management”, Dreamtech Press	

OPTIMIZATION TECHNIQUES (SET/CS/BT/OE503)		
Course Objective	1. Introduce students to classical optimization techniques and their applications in solving engineering and management problems. 2. Cover various optimization problem formulations, including single-variable and multi-variable optimization with and without constraints. 3. Explore linear programming concepts, including the simplex method and transportation problems, and their applications in real-world scenarios. 4. Introduce dynamic programming and its application in solving multistage decision processes efficiently. 5. Discuss simulation modeling as a powerful tool for analyzing complex systems and decision-making processes.	
Course Outcome	1. Develop an understanding of classical optimization techniques and their relevance in solving a wide range of optimization problems. 2. Gain proficiency in formulating and solving optimization problems, including linear programming, transportation problems, and integer programming. 3. Acquire skills in using optimization techniques to solve real-world engineering and management problems efficiently. 4. Demonstrate the ability to apply dynamic programming principles to solve multistage decision problems and integer programming problems. 5. Understand the principles of simulation modeling and its applications, along with the advantages and limitations of simulation techniques.	
Module Name	**Content**	**No. of Hrs.**
Introduction to Classical Optimization Techniques	Introduction to Classical Optimization Techniques: Statement of an Optimization problem – design vector – design constraints – constraint surface – objective function – objective function surfaces – classification of Optimization problems. Classical Optimization Techniques: Single variable Optimization, Multi variable Optimization with and without constraints, Multivariable Optimization with equality constraints - solution by method of Lagrange multipliers, Multivariable Optimization with inequality constraints - Kuhn – Tucker conditions.	10
Linear Programming	Various definitions, statements of basic theorems and properties, Advantages, Limitations and Application areas of Linear Programming, Graphical method of Linear Programming problem. Simplex Method – Phase me and Phase II of the Simplex Method, The Revised Simplex method, Primal and Dual Simplex Method, Big –M method.	8
Transportation Problem	Finding initial basic feasible solution by north – west corner rule, least cost method and Vogel's approximation method – testing for optimality of balanced transportation problems. (Including assignment and travelling salesman problems) (No degeneracy problems). Queuing Models : Essential features of queuing systems, operating characteristics of queuing system, probability distribution in queuing systems, classification of queuing models, solution of queuing M/M/1 : ¥ /FCFS,M/M/1 : N/FCFS, M/M/C : ¥/FCFS, M/M/C : N/FCFS.	8
Dynamic Programming	Dynamic programming multistage decision processes – types – concept of sub optimization and the principle of optimality – computational procedure in dynamic programming – examples illustrating the calculus method of solution - examples illustrating the tabular method of solution. Integer Programming: Pure and mixed integer programming problems, Solution of Integer programming problems – Gomory's all integer cutting plane method and mixed integer method, branch and bound method, Zero-one programming.	8
Simulation Modeling	Introduction, Definition and types, Limitations, Various phases of modeling, Monte Carlo method, Applications, advantages and limitations of simulation.	8
	Total No. of Hours	**42**
Text books	1. S.S.Rao, "Engineering optimization: Theory and practice", New Age International (P) Limited. 2. H A Taha, "Operations Research: An Introduction", 5th Edition, Macmillan, New York.	
References	1. K.V. Mittal and C. Mohan, "Optimization Methods in Operations Research and systems Analysis", International (P) Limited, Publishers. 2. G. Hadley, "Linear programming", Narosa Publishing House, New Delhi. 3. NVR Naidu, G Rajendra, T Krishna Rao, "Operations Research", I K International Publishing house, New Delhi.	

Semester VI

S. No.	Category	Course Code	Course Title	L	T	P	Contact Hrs./Week	Credits
1	Core Subjects	SET/CS/BT/C601	Compiler Design	3	1	-	4	4
2		SET/CS/BT/C602	Computer Graphics	3	1	-	4	4
3		SET/CS/BT/C603	Cryptography and Network Security	3	1	-	4	4
4			Program Elective-2	3	1	-	4	4
5	Open Elective/Inter-disciplinary Subject		Open Elective-2	3	1		4	4
6	Core Subjects Based Labs	SET/CS/BT/C604	Compiler Designing Lab	-		1	3	1
7		SET/CS/BT/C605	Computer Graphics Lab			1	3	1
8	Communication skills/CC	SET/CS/BT/M606	Communication skill Course* /Technical Seminar	-	-	1	4	2
9	Skill Course	SET/CS/SC/C607	Mini Project			1	4	2
Total				15	5	4	34	26

* University will prepare communication courses in Modern/Indian languages from which student will select one language course. The course will be more on applied side with giving students a chance to develop their soft skills.

Program Elective and Open Elective Courses

S. No.	Category	Course Code	Course Title	L	T	P	Contact Hrs./Week	Credits
1	**Program Elective-2**	SET/CS/BT/E601	Data Mining and Data Warehousing	3	1	-	4	4
2		SET/CS/BT/E602	E-Commerce	3	1	-	4	4
3		SET/CS/BT/E603	Data Science	3	1	-	4	4
1	**Open Elective-2**	SET/CS/BT/OE601	Robotic Engineering	3	1	-	4	4
2		SET/CS/BT/OE602	Web Technology	3	1		4	4
3		SET/CS/BT/OE603	Digital Image Processing	3	1		4	4

Note: Entry/Exit Option with Advanced Diploma/ Bachelor degree in Vocational Education (Computer Science and Engineering), may be allowed after completion of **SSD course work** in any one semester within one to six semesters and successfully completing three years, and as per requirements defined in the admission procedure and eligibility of the School.

COMPILER CONSTRUCTION (SET/CS/BT/C601)

Course Objective	1. Provide an introduction to compilers and their role in translating high-level programming languages into machine code. 2. Cover the phases and passes involved in the compilation process, including lexical analysis, syntax analysis, semantic analysis, code generation, and optimization. 3. Introduce finite state machines, regular expressions, and formal grammars, along with their applications to lexical and syntax analysis. 4. Explore syntactic specification of programming languages using context-free grammars and parsing techniques such as shift-reduce parsing and operator precedence parsing. 5. Discuss syntax-directed translation schemes, symbol tables, run-time administration, error detection, recovery, and basic concepts of code optimization.
Course Outcome	1. Gain an understanding of the compilation process and the various phases involved, enabling the development of basic compilers. 2. Acquire proficiency in implementing lexical analyzers using finite state machines and regular expressions. 3. Develop skills in specifying the syntax of programming languages using context-free grammars and constructing parsers for efficient syntax analysis. 4. Demonstrate the ability to implement syntax-directed translators and handle symbol tables for managing scope information. **5.** Understand the basics of error detection and recovery in the compilation process and introduce the concepts of code optimization for improving program efficiency.

Module Name	Content	No. of Hrs.
Fundamental Concept	Introduction to Compiler, Phases and passes, Bootstrapping, Finite state machines and regular expressions and their applications to lexical analysis, Implementation of lexical analyzers, lexical-analyzer generator, LEX-compiler, Formal grammars and their application to syntax analysis, BNF notation, ambiguity, YACC.	10
Syntactic specification of programming languages	Context free grammars, derivation and parse trees, capabilities of CFG. Basic Parsing Techniques, Parsers, Shift reduce parsing, operator precedence parsing, top down parsing, predictive parsers Automatic Construction of efficient Parsers: LR parsers, the canonical Collection of LR(0) items, constructing SLR parsing tables, constructing Canonical LR parsing tables, Constructing LALR parsing tables, using ambiguous grammars, an automatic parser generator, Implementation of LR parsing tables, constructing LALR sets of items.	12
Syntax-directed Translation	Syntax-directed Translation schemes, Implementation of Syntax- directed Translators, Intermediate code, postfix notation, Parse trees & syntax trees, three address code, quadruple & triples, translation of assignment statements, Boolean expressions, statements that alter the flow of control, postfix translation, translation with a top down parser. More about translation: Array references in arithmetic expressions, procedures call, declarations, and case statements.	10
Symbol Tables	Data structure for symbols tables, representing scope information. Run-Time Administration: Implementation of simple stack allocation scheme, storage allocation in block structured language. Error Detection & Recovery: Lexical Phase errors, syntactic phase errors semantic errors. Introduction to code optimization: Loop optimization, the DAG representation of basic blocks, value Numbers and algebraic laws, Global Data-Flow analysis.	10
	Total No. of Hours	**42**

Textbooks	1. Aho, Sethi and Ullman, "Compiler Design", AddisionWesley.
References	1. Introduction to Automata Theory, Languages, and Computation, Johne E. Hopcroft, Rajeev Motwani, Jeffrey D. Ulman, Pearson Education 2. Advanced Compiler Design and Implementation, Steven Muchnick, Morgan Kaufman Publication

COMPUTER GRAPHICS (SET/CS/BT/C602)

Course Objective	1. Provide an understanding of graphics primitives and their representation on display devices, including points, lines, polygons, and segments. 2. Cover the generation and manipulation of graphics primitives, including line generation, polygon filling, and segment manipulation. 3. Introduce transformations, windowing, and clipping techniques for manipulating and viewing graphics objects in two and three dimensions. 4. Explore interaction techniques for handling hardware input devices and implementing interactive graphics applications. 5. Discuss hidden line and surface removal algorithms, as well as rendering and illumination techniques for producing realistic graphics.
Course Outcome	1. Develop proficiency in generating and manipulating graphics primitives on display devices, enabling the creation of basic graphical applications. 2. Acquire skills in implementing line generation algorithms, polygon filling algorithms, and segment manipulation techniques for efficient graphics rendering. 3. Demonstrate the ability to apply transformations, windowing, and clipping techniques to manipulate and view graphics objects effectively. 4. Gain practical experience in implementing interactive graphics applications, including handling input devices and implementing interactive techniques. 5. Understand and implement advanced graphics algorithms such as hidden line and surface removal, as well as rendering and illumination techniques, to produce visually appealing and realistic graphics.

Module Name	**Content**	**No. of Hrs.**
Graphics Primitives	Display devices, Primitive devices, Display File Structure, Display control Text.	4
Line generation, Polygon, Segments	Points lines, Planes, Pixels and Frame buffers, vector and character generation. Polygon Representation, Entering polygons, Filling polygons. Segments table, creating deleting and renaming segments, visibility, image transformations.	12
Transformations, Windowing and Clipping	Matrices transformation, transformation routines, displays procedure. Viewing transformation and clipping, generalize clipping, multiple windowing. Three Dimension: 3-D geometry primitives, transformations, Projection clipping.	12
Interaction	Hardware input devices handling algorithms, Event handling echoing, Interactive techniques.	6
Hidden Line and Surface	Back face removal algorithms, hidden line methods.	4
Rendering and Illumination	Introduction to curve generation, Bezier, Hermite and Bspline algorithms And their comparisons.	4
	Total No. of Hours	**42**

Textbooks	1. Rogers, "Procedural Elements of Computer Graphics", McGrawHill 2. Asthana, Sinha, "Computer Graphics", Addison Wesley Newman and Sproul, "Principle of Interactive Computer Graphics", McGrawHill.
References	3. Steven Harrington, "Computer Graphics", A Programming Approach, 2ndEdition 4. Rogar and Adams, "Mathematical Elements of Computer Graphics", McGrawHill.

CRYPTOGRAPHY AND NETWORK SECURITY (SET/CS/BT/C 603)		
Course Objective	1. Provide a comprehensive understanding of security attacks, services, and mechanisms in computer systems. 2. Cover classical encryption techniques and modern block ciphers, including their principles, strengths, and modes of operation. 3. Introduce mathematical concepts such as prime numbers, modular arithmetic, and discrete logarithms, essential for understanding cryptographic algorithms. 4. Explore public key cryptosystems, including the RSA algorithm, Diffie-Hellman key exchange, and elliptic curve cryptography. 5. Discuss message authentication, hash functions, digital signatures, and authentication applications, including Kerberos, X.509, and digital certificate standards.	
Course Outcome	1. Develop proficiency in analyzing security attacks and mechanisms, enabling the design of secure computer systems. 2. Acquire skills in implementing classical encryption techniques and understanding the principles and strengths of modern block ciphers. 3. Understand mathematical concepts essential for cryptography, including prime numbers, modular arithmetic, and discrete logarithms. 4. Demonstrate the ability to implement public key cryptosystems and understand their security properties and key management protocols. 5. Gain practical experience in implementing message authentication, hash functions, digital signatures, and authentication applications, facilitating secure communication and data integrity in computer systems.	
Module Name	**Content**	**No. of Hrs.**
Module 1	Security attacks, Services and Mechanism, Conventional encryption model, classical encryption techniques substitution ciphers and transposition ciphers, cryptanalysis, stereography, stream and block ciphers. Modern Block Ciphers: Block ciphers principals, Shannon's theory of confusion and diffusion, fiestal structure, data encryption standard (DES), strength of DES, differential and linear crypt analysis of DES, block cipher modes of operations, triple DES, IDEA encryption and decryption, strength of IDEA, confidentiality using conventional encryption, traffic confidentiality, key distribution, random number generation.	10
Module 2	Introduction to graph, ring and field, prime and relative prime numbers, modular arithmetic, Fermat's and Euler's theorem, primality testing, Euclid's Algorithm, Chinese Remainder theorem, discrete logarithms. Principals of public key crypto systems, RSA algorithm, security of RSA, key management, Diffle-Hellman key exchange algorithm, introductory idea of Elliptic curve cryptography, Elganel encryption.	8
Module 3	Message Authentication and Hash Function: Authentication requirements, authentication functions, message authentication code, hash functions, birthday attacks, security of hash functions and MACS, MD5 message digest algorithm, Secure hash algorithm(SHA). Digital Signatures: Digital Signatures, authentication protocols, digital signature standards (DSS),proof Of digital signature algorithm.	8
Module 4	Authentication Applications: Kerberos and X.509, directory authentication service, electronic mail security-pretty good privacy (PGP), S/MIME.	8
Module 5	IP Security: Architecture, Authentication header, Encapsulating security payloads, combining security associations, key management. Web Security: Secure socket layer and transport layer security, secure electronic transaction (SET). System Security: Intruders, Viruses and related threads, firewall design principals, trusted systems.	8
	Total No. of Hours	**42**
Textbooks	1. William Stallings, "Cryptography and Network Security: Principals and Practice", Prentice Hall, New Jersy.	
References	1. Johannes A. Buchmann, "Introduction to Cryptography",Springer-Verlag. 2. B. Forouzan, "Cryptography and Network Security, TMH	

Program Elective Courses

DATA MINING AND DATA WAREHOUSING (SET/CS/BT/E 601)		
Course Objective	1. Provide an introduction to data mining and data warehouses, including their fundamental concepts and architectures. 2. Cover data pre-processing techniques such as cleaning, integration, transformation, reduction, and discretization essential for preparing data for analysis. 3. Introduce association rule mining, classification, prediction, and cluster analysis techniques for extracting meaningful patterns and insights from large datasets. 4. Explore various data mining primitives, query languages, graphical user interfaces, and architectures used in data mining applications. 5. Discuss recent trends and applications of data mining, including multidimensional analysis, complex data object mining, spatial databases, multimedia databases, and text databases.	
Course Outcome	1. Develop a comprehensive understanding of data mining and data warehousing concepts, enabling the design and implementation of effective data mining solutions. 2. Acquire skills in data preprocessing techniques for cleaning, integrating, transforming, and reducing data noise and redundancy. 3. Demonstrate proficiency in association rule mining techniques, including single-dimensional and multidimensional association rule mining, and correlation analysis. 4. Understand classification, prediction, and cluster analysis methods, including decision tree induction, Bayesian classification, and partitioning methods. 5. Gain practical experience in applying data mining techniques to various real-world applications, including spatial databases, multimedia databases, time series data, and text databases, contributing to informed decision-making and trend analysis.	
Module Name	**Content**	**No. of Hrs.**
Fundamental of Data Mining, and Data Warehouses	Introduction, Data Warehouse, Multidimensional Data Model, Data Warehouse Architecture, Implementation - Data Warehousing to Data Mining –Data warehousing components-building a data warehouse – mapping the data warehouse to an architecture - data extraction - cleanup- transformation tools- metadata – OLAP - Patterns and models - Data visualization principles.	8
Data Preprocessing, Language, Architectures, Concept Description	Preprocessing, Cleaning, Integration, Transformation, Reduction, Discretization, Concept Hierarchy Generation, Data Mining Primitives, Query Language, Graphical User Interfaces, Architectures, Concept Description, Data Generalization, Characterizations, Class Comparisons, Descriptive Statistical Measures	8
Association Rule	Association Rule Mining, Single-Dimensional Boolean Association Rules from Transactional Databases, Multi-Level Association Rules from Transaction Databases-mining multidimensional Association rules –association mining to correlation analysis-constraint based association mining.	8
Classification and Prediction	Classification and Prediction, Issues, Decision Tree Induction, Bayesian Classification, Association Rule Based, Other Classification Methods, Prediction, Classifier Accuracy..	8
Cluster Analysis	Cluster Analysis, Types of data, Categorization of methods, Partitioning methods, hierarchical methods, density based methods, grid based methods - Outlier Analysis. Recent trends - Multidimensional Analysis and Descriptive Mining of Complex Data Objects, Spatial Databases, Multimedia Databases, Time Series and Sequence Data, Text Databases, World Wide Web, Applications and Trends in Data Mining	10
	Total No. of Hours	**42**

Textbooks	1. Alex Berson and Stephen J. Smith, “Data Warehousing, Data mining and OLAP”, Tata McGraw-Hill, 2004. (UNIT V)
References	1. Margaret H.Dunham, “Data Mining: Introductory and Advanced Topics”, Pearson Education, 2004. 2.Sam Anahory and Dennis Murry, “Data Warehousing in the Real World”, PearsonEducation, 2003. 3. J. Han and M. Kamber, “Data Mining: Concepts and Techniques”,Harcourt India /Morgan Kauffman, 2001.

E-COMMERCE (SET/CS/BT/E602)		
Course Objective	1. Explore the economic potential of electronic commerce (E-commerce) and the incentives driving businesses to engage in online transactions. 2. Examine the forces behind E-commerce and its advantages and disadvantages, along with its impact on traditional business models. 3. Understand the architectural framework of E-commerce systems and the network infrastructure supporting online transactions. 4. Discuss security issues in E-commerce, including the importance of firewalls, transaction security, and encryption techniques to protect sensitive information. 5. Explore electronic payment systems, including protocols like SET, payment gateways, digital tokens, and various methods of online payments.	
Course Outcome	1. Develop a comprehensive understanding of the economic potential and incentives driving E-commerce, enabling students to identify opportunities and challenges in online business. 2. Gain insight into the architectural framework of E-commerce systems, including the network infrastructure required for secure and efficient transactions. 3. Acquire knowledge of security issues in E-commerce and understand the importance of implementing security measures such as firewalls and encryption techniques to protect data integrity and confidentiality. 4. Demonstrate proficiency in evaluating different electronic payment systems and understanding their functionalities and security features. 5. Understand the application of E-commerce technologies in various business contexts, including online banking, electronic data interchange (EDI), and emerging trends in mobile commerce, preparing students for careers in digital business environments.	
Module Name	**Content**	**No. of Hrs.**
Technology and Prospects	Economic potential of electronic commerce, Incentives for engaging in electronic commerce, forces behind E-Commerce, Advantages and Disadvantages, Architectural framework, Impact of E-Commerce on business.	8
Network Infrastructure of E- Commerce	Internet and Intranet based E-Commerce Issues, problems and prospects, Network Infrastructure, Network Access Equipment's, Broadband telecommunication (ATM, ISDN, and FRAME RELAY). Mobile Commerce: Introduction, Wireless Application Protocol, WAP Technology, Mobile Information device, Mobile Computing Applications.	10
Web Security	Security Issues on web, Importance of Firewall, components of Firewall, Transaction security, Emerging client server, Security Threats, Network Security, Factors to consider in Firewall design, Limitation of Firewalls.	8
Encryption	Encryption techniques, Symmetric Encryption-Keys and data encryption standard, Triple encryption. Asymmetric encryption-Secret key encryption, public and private pair key encryption, Digital Signature, Virtual Private Network.	8
Electronic Payments	Overview, The SET protocol, payment Gateway, certificate, digital Tokens, Smart card, credit card, magnetic strip card, E-Checks, Credit/Debit card based EPS, online Banking EDI Application in business	8
	Total No. of Hours	**42**
Textbooks	1. E-Commerce, Ritendra Goel, New Age International Publishers	
References	1. RaviKalakota,AndrewWinston,:FrontiersofElectronicCommerce"Addison Wesley. 2. Bajaj and Nag. "E-Commerce the cutting edge of Business".TMH. 3. P. Loshin, John Vacca, "Electronic Commerce" Firewall Media,N.Delhi. 4. E Business & Commerce: BrahmCazner, Wileydreamtech.	

DATA SCIENCE (SET/CS/BT/E603)		
Course Objective	1. Provide an overview of data science, including its applications, tools, and techniques, along with programming fundamentals in languages such as Python and R. 2. Introduce mathematical concepts essential for data science, including calculus, linear algebra, probability, and statistics, enabling students to understand and analyse data effectively. 3. Explore machine learning concepts and algorithms, including supervised, unsupervised, and reinforcement learning, preparing students to apply machine learning techniques to real-world problems. 4. Cover big data technologies such as Hadoop, Spark, and NoSQL databases, along with deep learning concepts and algorithms, enabling students to work with large datasets and complex models. 5. Discuss data visualization techniques, tools, and database management systems, including relational database concepts, SQL querying, database design, and data manipulation techniques, enhancing students' ability to visualize and manage data efficiently.	
Course Outcome	1. Develop a comprehensive understanding of data science principles, techniques, and tools, enabling students to analyse and interpret data effectively. 2. Acquire proficiency in programming languages such as Python and R, along with data structures, algorithms, and object-oriented programming principles, facilitating software development in data science projects. 3. Gain knowledge of mathematical concepts necessary for data science, including calculus, linear algebra, probability, and statistics, enhancing students' ability to perform advanced data analysis. 4. Demonstrate proficiency in applying machine learning algorithms to solve real-world problems, including regression, classification, clustering, and neural networks, fostering a deeper understanding of machine learning concepts. 5. Understand big data technologies, deep learning concepts, data visualization techniques, and database management systems, enabling students to work with large datasets, complex models, and visualization tools effectively in data science projects.	
Module	**Content**	**No. of Hrs.**
Introduction to Data Science and Programming Fundamentals	Overview of Data Science, Applications of Data Science, Need of Data Science, Tools and Technique in Data Science, Basics of Programming Language (Python, R), Data Structures, Algorithms, Object-Oriented Programming Principles.	8
Mathematics for Data Science	Calculus, Linear Algebra, Probability, Statistics, Regression Analysis, Hypothesis Testing.	8
Machine Learning	Introduction to Machine Learning, Supervised Learning, Unsupervised Learning, Reinforcement Learning, Algorithms (Linear Regression, Logistic Regression, Decision Tree, Support Vector Machines, Clustering, Neural Networks.	8
Big Data Technologies and Deep Learning	Introduction to Big Data, Big Data Technologies (Hadoop, Spark, NoSQL, databases), Introduction to Deep Learning, Deep Neural Network, Convolutional Neural Network, Recurrent Neural Network, Applications in Image Recognition, Speech Recognition, and Natural Language Processing.	10
Data Visualization and Database Management Systems	Overview of Data Visualization, Techniques of Data Visualization, Data Visualization Tools, Introduction to Database Systems, Relational Database Concepts, SQL Querying, Database Design, Data Manipulation Techniques.	10
	Total No. of Hours	44
Text Books	**1.** "Data Science and Machine Learning using Python" By Dr Reema Thareja , McGraw Hill Publication	
References	1. "Introduction to Data Science: Practical Approach with R and Python" by B. Uma Maheswari, R. Sujatha, Wiley Publication 2. "INTRODUCTION TO DATA SCIENCE" by Dr. Sushil Dohare, Dr. V SelvaKumar, Sachin Raval, Dr. Sumegh Shrikant Tharewal, Published by Xoffencer	

Open Elective Courses

ROBOTIC ENGINEERING(SET/CS/OE601)		
Course Objective	1. Introduce students to the field of robotics, including its history, types of robots, and various applications in different industries. 2. Provide an understanding of robot kinematics, covering coordinate systems, forward and inverse kinematics, and the Jacobian matrix. 3. Explore robot dynamics and control, including Newton-Euler equations, Lagrange equations, dynamic models of robots, and different control techniques such as PID control and feedback control. 4. Discuss sensors and perception in robotics, including types of sensors, vision systems, localization, mapping, and actuators such as DC motors, servo motors, grippers, and end-effectors. 5. Cover robot programming aspects, including programming languages commonly used in robotics such as C++ and Python, robot operating systems, simulation, and modelling techniques.	
Course Outcome	1. Gain knowledge about the history, types, and applications of robotics, enabling students to understand the scope and potential of robotics in various fields. 2. Understand the principles of robot kinematics, including coordinate systems and kinematic equations, facilitating the analysis and design of robot manipulators. 3. Acquire proficiency in robot dynamics and control, allowing students to model robot dynamics and implement control strategies for robot motion. 4. Learn about sensors and perception systems used in robotics, along with actuators for manipulation tasks, enabling students to design and implement robotic systems with sensing and actuation capabilities. 5. Develop skills in robot programming, including using programming languages and robot operating systems, facilitating the development and deployment of robotic applications in real-world scenarios.	
MODULE	**CONTENT**	**No. of Hrs.**
Introduction to Robotics and Robot Kinematics	Introduction to Robotics, History of Robotics, Types of Robots, Application of Robotics, Robot Kinematics: Coordinate Systems, Forward and Inverse Kinematics, Jacobian Matrix.	8
Robot Dynamics and Robot Control	Newton -Euler Equations, Lagrange Equations, Dynamics Models of Robots, Robot Controls: PID Control, Feedback Control, Motion Planning.	8
Sensor and Perceptions and Actuators and Manipulations	Introduction to Sensors, Types of Sensors, Range Sensors, Vision Systems, Localization and Mapping, Actuators: Introduction to Actuators, DC Motors, Servo Motors, Grippers and End-Effectors.	10
Robot Programming	Programming Language for Robotics (e.g., C++, Python), Robot Operating Systems, Simulation and Modelling.	10
Robot Design and Manipulation	Design Consideration, Mechanical Design, Electrical and Electronic Components.	6
	Total No. of Hours	42
Text Books	1. Introduction to Robotics: Mechanics and Control by John J. Craig	
References	1. "Robot Modelling and Control" by Mark w. Spong, Seth Hutchinson, and M. Vidyasagar 2. "ROS Robotics" by Example by Carol Fairchild 3. "Robotics: Modelling, Planning and Control" by Bruno Siciliano, Lorenzo Sciavicco, Luigi Villani, and Giuseppe Orilo.	

WEB TECHNOLOGY (SET/CS/BT/OE602)		
Course Objective	1. Provide an introduction to web technologies, covering the history and evolution of the World Wide Web and the basics of client-server architecture. 2. Teach HTML (Hypertext Mark-up Language) and CSS (Cascading Style Sheets), including the structure of HTML documents, semantic mark-up, forms, multimedia embedding, styling HTML elements, and layout techniques. 3. Introduce JavaScript programming fundamentals, focusing on DOM manipulation, event handling, and asynchronous programming using Promises and a sync/await. 4. Explore web development tools such as text editors, IDEs, version control with Git, and browser developer tools to aid in web development. 5. Cover backend technologies and databases, including server-side programming languages like Node.js, Python, and PHP, server-side frameworks, relational and NoSQL databases, and database design principles.	
Course Outcome	1. Understand the structure and components of web development, including HTML, CSS, and JavaScript, enabling students to create and style web pages. 2. Gain proficiency in client-side scripting with JavaScript, allowing students to manipulate the Document Object Model (DOM) and handle events on web pages. 3. Familiarize with common web development tools and techniques, empowering students to efficiently develop and debug web applications. 4. Acquire knowledge of backend technologies and databases, enabling students to build dynamic web applications with server-side logic and interact with databases. 5. Learn about web security best practices and performance optimization techniques, equipping students to develop secure, high-performing web applications.	
MODULE	**CONTENT**	**No. of Hrs**
Introduction to Web Technologies:	Introduction to Website, History and evolution of the World Wide Web, Client-server architecture, Basic terminology (HTML, CSS, JavaScript, etc.).	8
HTML (Hypertext Mark-up Language) and CSS (Cascading Style Sheets)	Structure of HTML documents, Semantic mark-up, Forms and input elements, Multimedia embedding. CSS: Styling HTML elements, Layout techniques (flexbox, grid), Responsive design, CSS frameworks (Bootstrap, Foundation).	10
JavaScript and Web Development Tools	Fundamentals of JavaScript programming, DOM manipulation, Event handling, Asynchronous programming with Promises and a sync/await, Web Development Tools: Text editors and IDEs, Version control with Git, Browser developer tools.	10
Backend Technologies and Databases	Introduction to server-side programming languages (e.g., Node.js, Python, PHP), Server-side frameworks (e.g., Express.js, Flask, Django), Databases: Relational databases (e.g., MySQL, PostgreSQL), NoSQL databases (e.g., MongoDB), Database design and normalization.	10
Web Security and Web Performance Optimization	Common security threats (e.g., XSS, CSRF), Authentication and authorization, HTTPS and SSL/TLS, Web Performance Optimization: Minification and compression, Caching strategies, Load balancing and content delivery networks (CDNs).	8
	Total No. of Hours	46
Text Books	1. Web Programming, building internet applications, Chris Bates 2nd edition, WILEY Dreamtech 2. Core SERVLETS ANDJAVASERVER PAGES VOLUME 1: CORE TECHNOLOGIES By Marty Hall and Larry Brown Pearson	
References	1. . An Introduction to WEB Design and Programming –Wang-Thomson 2. 2. PHP: The Complete Reference Steven Holzner TataMcGraw-Hill.	

DIGITAL IMAGE PROCESSING (SET/CS/BT/OE603)		
Course Objective	1. Provide an introduction to image processing, covering digital images, image sampling, and multiview geometry. 2. Teach image enhancement techniques, including spatial and frequency domain enhancements, to improve image quality and clarity. 3. Explore image restoration methods to remove noise and improve the quality of degraded images. 4. Introduce color image processing fundamentals, color models, and color transformation techniques. 5. Cover advanced topics such as wavelets, morphological image processing, image compression, and segmentation algorithms.	
Course Outcome	1. Understand the principles and fundamentals of digital image processing, including image representation, sampling, and basic operations. 2. Gain proficiency in applying spatial and frequency domain techniques for image enhancement, restoration, and color manipulation. 3. Learn advanced image processing techniques such as wavelet transforms and morphological operations for feature extraction and analysis. 4. Acquire knowledge of image compression algorithms and segmentation methods for efficient storage and analysis of image data. 5. Develop skills to implement and apply various image processing algorithms to solve real-world problems in fields like computer vision, medical imaging, and remote sensing.	
Module Name	**Content**	**No. of Hrs.**
Introduction and Fundamentals	Introduction to Image Processing, Digital Images- The Eye, Brightness, Image Sampling, Neighbors of Pixels Distance. Multiview Geometry- Stereo Vision, The correspondence problem; Algorithms for Stereo Matching,	8
Image Enhancement	Spatial Image Enhancements- Transformations: Negative, Log, Power, Histogram, Subtraction, Averaging, Smoothing, Laplacian. Frequency Domain Image Enhancements 1D FT(Fourier Transform), Inverse, 2D FT, Filtering, Low pass, High pass, Un-sharp, High-Boost, Use of FT, Fast FT	8
Image Restoration Color Image Processing	Noise, Mean filter, Median, Min, Max, Midpoint, and Adaptive filters, Frequency Domain, etc... Color Fundamentals, Color Models, Converting Colors to Different Models, Color Transformation, Smoothing and Sharpening, Color Segmentation.	8
Applications of Wavelets Morphological Image Processing	Multi Resolution Expansions, Wavelet Transform in One Dimension, The Fast Wavelet Transform, Wavelet Transform in Two Dimensions Erosion and Dilation, Opening and Closing, The Hit or Miss Transformations, Some Basic Morphological algorithms	8
Image Compression And Segmentation	Need for Data Compression, Huffman Coding, Golomb coding, Arithmetic coding, LZW coding, Run Length coding, Bit plane coding and Wavelet coding Edge Detection, Thresholding, Region based Segmentation, Segmentation using Morphological Watersheds and the use of motion in Segmentation. Algorithm.	10
	Total No. of Hours	**42**
Textbooks	1. Rafael C. Gonzalvez and Richard E. Woods, "Digital Image Processing", 2nd Edition, Pearson Education.	
References	1. R.J. Schalkoff. "Digital Image Processing and Computer Vision", Wiley 2. A.K. Jain, "Fundamentals of Digital Image Processing", Prentice Hall 3. D. A. Forsyth , J. Ponce, "Computer Vision: A Modern Approach", Prentice Hall.	

Semester VII

S. No.	**Category**	**Course Code**	**Course Title**	L	T	P	Contact Hrs./Week	Credits
1	Core Subjects	SET/CS/BT/C701	Artificial Intelligence	3	1	-	4	4
2			Program Elective-3	3	1	-	4	4
3			Program Elective-4	3	1	-	4	4
4	Core Subjects Based Labs/Industrial Oriented Training	SET/CS/BT/C702	Artificial Intelligence Lab	-	-	1	3	1
5		SET/CS/BT/S703	Industrial Training Seminar	-	-	1	3	1
6	Life Skills and personality development		Essential Management Practices*	2	-	-	2	2
7	Skill Course	SET/CS/SC/C704	Project Stage-1	-	-	1	8	4
Total				11	3	3	28	20

***** University will prepare a course with focus on essential management practices.

S. No.	**Category**	**Course Code**	**Course Title**	L	T	P	Contact Hrs./Week	Credits
1	**Program Elective-3**	SET/CS/BT/E701	Wireless and Mobile Computing	3	1	-	4	4
2		SET/CS/BT/E702	Security Architecture & Operating System Security	3	1	-	4	4
3		SET/CS/BT/E703	Neural Network	3	1	-	4	4
1	**Program Elective-4**	SET/CS/BT/E704	Real Time System	3	1	-	4	4
2		SET/CS/BT/E705	Cloud Computing	3	1	-	4	4
3		SET/CS/BT/E706	Computer Vision	3	1	-	4	4

ARTIFICIAL INTELLIGENCE (SET/CS/BT/C701)		
Course Objective	1. Understand Problem Solving Methods: Introduce various problem-solving techniques, including production systems, state space search, and control strategies. 2. Explore Heuristic Search Techniques: Teach heuristic search methods such as hill climbing, breadth-first search, depth-first search, and best search. 3. Knowledge Representation: Cover different knowledge representation methods, including predicate logic, semantic nets, and fuzzy logic. 4. AI Applications: Provide an overview of AI applications in neural networks, natural language processing, speech recognition, and robotics. 5. Expert Systems Design: Explain the structure and design of expert systems, including their interaction with experts and practical application.	
Course Outcome	1. Proficient in Problem Solving Techniques: Gain proficiency in applying various problem-solving methods and control strategies to complex problems. 2. Heuristic Search Skills: Develop skills in implementing and utilizing heuristic search techniques to find optimal solutions efficiently. 3. Knowledge Representation Competency: Understand and apply different knowledge representation methods for effective AI problem-solving. 4. AI Application Insights: Acquire knowledge of AI applications and their practical uses in fields such as neural networks, natural language understanding, and robotics. **5.** Expert System Design: Learn how to design and implement expert systems, including understanding their structure, interaction, and applications in real-world scenarios.	
Module Name	**Content**	**No. of Hrs.**
Problem Solving Methods	Production systems ,State space search , Control strategies , Heuristic search, Forward and backward reasoning, Hill Climbing techniques, Breadth first search, Depth first search, Best search , Staged search., Predicate logic, Resolutionquestionanswering,Nonmonoticreasoning,Stasticalandprobalistic reasoning,	12
Knowledge Representation	Predicate logic, Resolution question answering, Nonmonoticreasoning , Stastical and probalistic reasoning, Fuzzy logic, Semantic Nets, Conceptual dependency, Frames, Scripts.	10
AI Application	Neural Networks, Natural language understanding, speech recognition and understanding, Learning, perception, AI robotics, satellite imaging and medical diagnosis.	10
Expert Systems	Structure of an expert system, interaction with an expert, Design of an expert system.	10
	Total No. of Hours	**42**
Textbooks	1. E. Rich & K. Knight : Artificial Intelligence.	
References	1. N. J. Nilsson : Principles of Artificial Intelligence 2. A. Barr, E. A. Fergenbaumand& P. R. Cohen : Artificial Intelligence. 3. 4. D. A. Waterman : A guide to Expert System.	

ARTIFICIAL INTELLIGENCE LAB (SET/CS/BT/C702)		
Lab Objective	1. To understand and implement fundamental algorithms in computer science using Python. 2. To develop problem-solving skills through coding classic algorithmic problems. 3. To gain proficiency in Python programming and improve coding practices. 4. To apply theoretical knowledge to practical coding challenges, enhancing comprehension of algorithms. 5. To foster logical thinking and algorithmic design skills applicable to real-world problems.	
Lab Outcome	1. Students will be able to implement and understand Breadth First Search and Depth First Search algorithms in Python. 2. Students will successfully code and execute classic problems such as the 8-Puzzle, Water-Jug, and Tic-Tac-Toe, demonstrating applied problem-solving skills. 3. Students will be proficient in implementing complex algorithms like the Travelling Salesman Problem and Alpha-Beta Pruning in Python. 4. Students will gain the ability to design and implement solutions for combinatorial problems such as the 8-Queens Problem and Tower of Hanoi. 5. Students will develop and enhance their logical reasoning and coding skills, preparing them for more advanced computational challenges and real-world applications.	
Module Name	**Content**	**No. of Hrs.**
	1. Write a Program to Implement Breadth First Search using Python. 2. Write a Program to Implement Depth First Search using Python. 3. Write a Program to Implement Tic-Tac-Toe game using Python. 4. Write a Program to implement 8-Puzzle problem using Python. 5. Write a Program to Implement Water-Jug problem using Python. 6. Write a Program to Implement Travelling Salesman Problem using Python. 7. Write a Program to Implement Tower of Hanoi using Python. 8. Write a Program to Implement Monkey Banana Problem using Python. 9. Write a Program to Implement Alpha-Beta Pruning using Python. 10. Write a Program to implement 8-Queens Problem using Python.	
Total No. of Hours		**36**

INDUSTRIAL TRAINING SEMINAR (SET/CS/BT/S703)		
Course Objective	1. To provide students with hands-on experience in industrial settings. 2. To enhance students' technical and soft skills. 3. To familiarize students with current industry practices and technologies. 4. To facilitate professional networking opportunities. 5. To develop students' critical thinking and problem-solving abilities.	
Course Outcome	1. Students will be able to apply theoretical concepts to practical situations. 2. Students will demonstrate improved technical proficiency with industry-standard tools and technologies. 3. Students will gain a better understanding of industry standards and expectations. 4. Students will develop enhanced communication skills through interactions and presentations. 5. Students will experience personal and professional growth, including increased confidence and adaptability.	
Module Name	**Content**	**No. of Hrs.**
	Student shall prepare a detailed report on her/his industrial training and deliver a seminar of 30 minutes.	-
	Total No. of Hours	-

PROJECT Stage - I (SET/CS/BT/S704)		
Course Objective	1. Literature Review and Research Skills: To enable students to conduct a thorough survey and study of published literature on a given topic, identifying key research papers, trends, and gaps. 2. Problem Identification and Definition: To teach students how to select and precisely define a relevant and feasible research problem within the scope of their assigned topic. 3. Preliminary Solution Development: To guide students in developing a preliminary approach to solving the identified problem, including conceptualizing models, simulations, experiments, and feasibility studies. 4. Analytical and Technical Proficiency: To enhance students' abilities to conduct preliminary analysis, modeling, or simulation, and to design experiments or feasibility studies related to their research problem. 5. Communication and Presentation Skills: To improve students' skills in preparing comprehensive written reports and delivering effective oral presentations to communicate their research findings and methodologies.	
Course Outcome	1. Enhanced Research Skills: Students will be proficient in conducting comprehensive literature reviews, identifying key research works, and understanding the current state of knowledge in their research area. 2. Clear Problem Definition: Students will be able to define a specific, relevant, and researchable problem within the context of their assigned topic. 3. Preliminary Solution Approaches: Students will develop the ability to conceptualize and formulate preliminary approaches to solving the identified problem, demonstrating initial feasibility and validity. 4. Technical Analysis and Design: Students will gain experience in performing preliminary analyses, designing models, simulations, or experiments, and evaluating the feasibility of their approaches. 5. Effective Communication: Students will be capable of preparing detailed written reports and delivering articulate oral presentations to effectively convey their research process, findings, and implications to a departmental committee.	
Module Name	**Content**	**No. of Hrs.**
	Project – I includes following assignments. • Survey and study of published literature on the assigned topic. • Select and define an appropriate problem. • Working out a preliminary approach to the Problem relating to the assigned topic. • Conducting Preliminary Analysis/ Modeling/ Experiment/ Simulation/ Experiment/ Design/Feasibility. • Preparing a Written Report on the Study conducted for presentation to the Department. • Final Seminar, as oral Presentation before a Departmental Committee.	**3x16**
	Total No. of Hours	**48**

Program Elective Courses

Wireless Network and Mobile Computing (SET/CS/BT/E701)		
Course Objective	1. To provide an in-depth understanding of the protocol architecture, physical layer, and sub-layers of cellular communication systems. 2. To educate students on the structure, services, and protocols of GSM, including localization, handover, and security mechanisms. 3. To familiarize students with Mobile IP, including packet delivery, handover management, and various mobile transport layer protocols. 4. To explore the concepts, challenges, and benefits of mobile computing, including ad hoc networks, and to understand wireless LAN protocols and Bluetooth technology. 5. To study the taxonomy, applications, and challenges of Mobile Ad Hoc Networks (MANETs) and examine various routing protocols.	
Course Outcome	1. Students will understand the detailed protocol architecture and physical layer of cellular communications, and differentiate between WLAN and Bluetooth technologies. 2. Students will gain comprehensive knowledge of GSM system architecture, including mobile services, radio interface, protocols, handover mechanisms, and security features. 3. Students will be proficient in the principles and functioning of Mobile IP, including packet delivery, handover management, and mobile-specific transport layer protocols. 4. Students will be able to explain the fundamental concepts, challenges, and benefits of mobile computing and ad hoc networks, including an understanding of wireless LAN and Bluetooth protocols. 5. Students will have an advanced understanding of MANETs, including their taxonomy, applications, and challenges, and will be able to implement and analyze various routing protocols such as DSR, AODV, DSDV, CBRP, and TORA.	
Module Name	**Content**	**No. of Hrs.**
Introduction to Cellular Communications	Protocol Architecture, Physical Layer, Channel Access Control Sub-layer, MAC Sub-layer, WLAN: Infrared vs. Radio Transmission, Infrastructure and Ad Hoc Networks, IEEE 802.11. Bluetooth.: User Scenarios, Physical Layer, MAC layer, Networking, Security, Link Management.	8
GSM	Mobile Services, System Architecture, Radio Interface, Protocols, Localization and calling, Handover, Security, and New Data Services	8
Mobile IP	lP and Mobile IP Network Layers, Packet Delivery and Handover Management, Location Management, Registration, Tunnelling and Encapsulation, Route Optimization, DHCP. Mobile Transport Layer: Conventional TCP/IP Protocols, Indirect TCP, Snooping TCP, Mobile TCP, Other Transport Layer Protocols for Mobile Networks.	8
Overview of Ad Hoc Networks:	Ad Hoc Networks, Challenges, and benefits of Mobile Computing, breakthrough Technology, Wireless Computing, Nomadic Computing, Mobile Computing, Ubiquitous Computing, Pervasive Computing, Invisible Computing, applications of mobile computing, Wireless and Mobile Computing Models, LAN Protocols: IEEE 802.11/a/g/n & Bluetooth, Data Management Issues. Sensor Networks- Challenges, Architecture, and Applications.	8
Mobile Ad hoc Networks (MANETs)	Taxonomy, Applications, Challenges in Mobile Environments, Hidden and exposed terminal problems, Routing Protocols- Proactive, Reactive, and Hybrid protocols, Dynamic State Routing (DSR), Ad hoc On-Demand Distance Vector (AODV), Destination Sequenced Distance – Vector Routing (DSDV), and Cluster Based Routing Protocol (CBRP), and Temporally Ordered Routing algorithm (TORA).	10
	Total No. of Hours	**42**
Textbooks	1. Jochen Schiller, “Mobile Communications”, Addison-Wesley, Second Edition, 2004 2. Charles E. Perkins, Ad hoc Networks, Addison Wesley, 2008.	
References	1. KazemSohraby, Daniel Minoli, TaiebZnati, Wireless Sensor Networks: Technology, Protocols, and Applications, Wiley, 2007. 2. Raj Kamal, “Mobile Computing”, Oxford University Press, 2007,	

SECURITY ARCHITECTURE & OPERATING SYSTEM SECURITY (SET/CS/BT/E702)		
Course Objective	1. To provide a foundational understanding of information systems, database management systems, and the architecture of information security. 2. To educate students on the security components of operating systems, user authentication methods, and email security protocols. 3. To develop skills in user administration, password policies, and managing privileges and roles within database systems. 4. To introduce various database application security models, including virtual private databases (VPD) and data encryption techniques. 5. To explore methods for auditing database activities and implementing privacy-preserving data mining techniques.	
Course Outcome	1. Students will gain a comprehensive understanding of database security, asset valuation, and security methods, and will be able to identify and mitigate vulnerabilities within an operating system. 2. Students will learn effective user administration practices, including creating, modifying, and managing user profiles, roles, and privileges, as well as designing and implementing robust password policies. 3. Students will be familiar with different types of application security models, understand the implementation of virtual private databases, and apply data encryption techniques to protect sensitive information. 4. Students will acquire the ability to audit database activities using various tools and techniques, understand the creation of DLL triggers, and perform security audits in both Oracle and SQL Server environments. 5. Students will develop knowledge of privacy-preserving data mining algorithms, including randomization methods, group-based anonymization, and distributed privacy-preserving techniques, and will understand their applications in real-world scenarios.	
Module Name	**Content**	**No. of Hrs.**
Introduction	Introduction-Information Systems- Database Management Systems-Information Security Architecture- Database Security–Asset Types and value-Security Methods Introduction-Operating System Overview-Security Environment – Components-Authentication Methods-User Administration-Password Policies-Vulnerabilities-E-mail Security	8
Administration of Users and Profiles, Password Policies, Privileges and Roles	Introduction-Authentication-Creating Users, SQL Server User-Removing, Modifying Users-Default, Remote Users-Database Links-Linked Servers-Remote Servers-Practices for Administrators and Managers-Best Practices Introduction-Defining and Using Profiles-Designing and Implementing Password Policies-Granting and Revoking User Privileges-Creating, Assigning and Revoking User Roles-Best Practices	10
Database Application Security Models andVirtual Private Databases	Introduction-Types of Users-Security Models- Application Types-Application Security Models-Data Encryption Introduction-Overview of VPD-Implementation of VPD using Views, Application Context in Oracle-Implementing Oracle VPD-Viewing VPD Policies and Application contexts using Data Dictionary, Policy Manager Implementing Row and Column level Security with SQL Server	8
Auditing Database Activities:	Using Oracle Database Activities-Creating DLL Triggers with Oracle-Auditing Database Activities with Oracle-Auditing Server Activity with SQL Server 2000-Security and Auditing Project Case Study	8
Privacy Preserving Data Mining Techniques	Introduction- Privacy Preserving Data Mining Algorithms-General Survey-Randomization Methods-Group Based Anonymization-Distributed Privacy Preserving Data Mining-Curse of Dimensionality-Application of Privacy Preserving Data Mining	6
	Total No. of Hours	**42**
Textbooks	1. Hassan A. Afyouni, “Database Security and Auditing”, Third Edition, CengageLearning, 2009. 2. Charu C. Aggarwal, Philip S Yu, “Privacy Preserving Data Mining”: Modelsand Algorithms, Kluwer Academic Publishers, 2008	
References	1. Ron Ben Natan, ”Implementing Database Security and Auditing”, ElsevierDigital Press, 2005	

NEURAL NETWORK (SET/CS/BT/E703)		
Course Objective	1. To provide a comprehensive understanding of the fundamental principles of neurocomputing and neuroscience, including neuron models and learning processes. 2. To introduce data processing techniques essential for neural network applications, such as scaling, normalization, transformation, and principal component analysis. 3. To explore the architecture and training algorithms of multilayer perceptrons and radial basis function (RBF) networks, including accelerated learning methods. 4. To study recurrent networks, self-organizing maps, and advanced techniques like principal component and independent component analysis with applications in image and signal processing. 5. To examine the complexity of neural network models and their integration with soft computing techniques such as neuro-fuzzy and genetic algorithms.	
Course Outcome	1. Students will understand the historical context and foundational concepts of neurocomputing and neuroscience, including knowledge representation and learning processes in neural networks. 2. Students will gain proficiency in various data processing methods, including transformations, principal component analysis, and eigenvector techniques, crucial for neural network performance. 3. Students will be able to design and implement multilayer perceptrons and RBF networks, apply backpropagation algorithms, and use heuristics to enhance algorithm efficiency. 4. Students will acquire skills in working with recurrent networks, self-organizing maps, and performing principal component and independent component analysis, with practical applications in image and signal processing. 5. Students will develop an understanding of the complexity of neural network models and learn to integrate these models with soft computing techniques for enhanced problem-solving capabilities in various domains.	
Module Name	**Content**	**No. of Hrs.**
Neuro computing and Neuroscience	Historical notes, human Brain, neuron Model, Knowledge representation, Al and NN. Learning process: Supervised and unsupervised learning, Error correction learning, competitive learning, adaptation, statistical nature of the learning process	8
Data processing	Scaling, normalization, Transformation (FT/FFT), principal component analysis, regression, co-variance matrix, eigen values & eigen vectors. Basic Models of Artificial neurons, activation Functions, aggregation function, single neuron computation, multilayer perceptron, least mean square algorithm, gradient descent rule, nonlinearly separable problems and bench mark problems in NN.	10
Multilayer Perceptions And RBF networks	Multilayered network architecture, back propagation algorithm, heuristics for making BP-algorithm perorms better. Accelerated learning BP (like recursive least square, quick prop, RPROP algorithm), approximation properties of RBF networks and comparison with multilayer perceptran.	8
Recurrent network	Recurrent network and temporal feed-forward network, implementation with BP, self-organizing map and SOM algorithm, properties of feature map and computer simulation. Principal component and Independent component analysis, application to image and signal processing.	8
Complexity of neural network	Complex valued NN and complex valued BP, analyticity of activation function, application in 2Dinformation processing. Complexity analysis of network models. Soft computing. Neuro-Fuzzy-genetic algorithm Integration.	8
	Total No. of Hours	**42**
Books	1. Kevin L. Priddy, Paul E. Keller – Artificial neural networks: An Introduction – SPIE Press, 2005 2. S. Raj sekaran, Vijayalakshmi Pari - Neural networks, Fuzzy logic and Genetic Algorithms	
References	3. G. J. Klir& T. A. Folger : Fuzzy sets, Uncertainty and Information. 4. Simon Haykin : Neural Networks. 5. B. Kosco : Neural Networks and Fuzzy systems: A Dynamical approach to Machine Intelligence. 6. J. Hertz &Korgh : Introduction to the Theory of Neural Computation	

REAL TIME SYSTEM (SET/CS/BT/E704)		
Course Objective	1. To provide an in-depth understanding of the characteristics and types of real-time tasks, including their timing constraints and scheduling algorithms. 2. To explore resource sharing, priority management, and handling task dependencies in real-time systems, particularly in multiprocessor and distributed environments. 3. To examine the features and functionalities of real-time operating systems (RTOS) and real-time databases, including design issues and concurrency control. 4. To study real-time communication requirements, models, and routing protocols in wide area networks (WANs) and local area networks (LANs). 5. To analyze the quality of service (QoS) requirements and protocols for real-time communications over packet-switched networks and ensure dependable real-time channels.	
Course Outcome	1. Students will understand the fundamental concepts of real-time scheduling, including clock-driven, event-driven, hybrid schedulers, and specific algorithms like EDF and RM scheduling. 2. Students will be able to manage resource sharing in real-time tasks, address priority inversion, implement priority inheritance and ceiling protocols, and handle task dependencies in multiprocessor and distributed systems. 3. Students will gain knowledge of the features and design issues of real-time operating systems and databases, and understand the characteristics and applications of temporal data. 4. Students will learn about the performance requirements, resource management, and routing protocols for real-time communication in wide area networks. 5. Students will develop the skills to implement bounded access protocols for LANs, manage real-time communications over packet-switched networks, and meet QoS requirements for reliable and efficient real-time communication.	
Module Name	**Content**	**No. of Hrs.**
Introduction	Characteristics – Types of Real-Time tasks – Timing constraints –Real-Time Scheduling - Basic concepts and classification of Algorithms – Clock-Driven Scheduling – Event-Driven Scheduling – Hybrid schedulers – EDF Scheduling – RM Scheduling and its Issues.	8
Resource Sharing and Dependencies among Real-Time Tasks	Resource sharing in Real Time tasks, Priority Inversion, Priority Inheritance Protocol, Highest Locker Protocol, Priority Ceiling Protocol, Handling Task dependencies – Scheduling Real-Time Tasks in Multiprocessor and Distributed Systems – Resource Reclaiming in Multiprocessor Real-Time Systems – Fault-Tolerant Task Scheduling in Multiprocessor Real-Time Systems.	10
Real-Time Operating System (RTOS)	Features of RTOS, Commercial Real-Time Operating Systems, Real-Time Databases - Applications, Design issues, Characteristics of Temporal Data, Concurrency control, Commercial Real-Time Databases	8
Real-Time Communication in Wide Area Networks	Introduction, Service and Traffic Models and Performance Requirements, Resource Management, Switching Subsystem, Route Selection in Real-Time Wide Area Networks - Basic Routing Algorithms, Routing during Real-Time Channel Establishment, Route Selection Approaches, Dependable Real-Time Channels	8
Real-Time Communication in a LAN:	Soft Real-Time Communication in a LAN – Hard Real-Time Communication in a LAN – Bounded Access Protocols for LANs – Real-Time Communications over Packet Switched Networks – QoS requirements – Routing and Multicasting	8
	Total No. of Hours	42
Textbooks	1. C. Siva Ram Murthy and G. Manimaran, “Resource Management in Real-Time Systems 2. Jane W.S. Liu, “Real-Time Systems”, Prentice Hall,	
References	1. Rajib Mall, “Real-Time Systems Theory and Practice”, Pearson Education 2. C.M. Krishna and Kang G. Shin, “Real-Time Systems”, McGraw-Hill International	

CLOUD COMPUTING (SET/CS/BT/E705)

Course Objective	1. To provide an understanding of the evolution, system models, and reference architecture of cloud computing, including insights into various service models (IaaS, PaaS, SaaS) and deployment models (public, private, hybrid). 2. To educate on virtualization technology, its definition, benefits, and implementation levels, as well as its critical role in cloud computing infrastructure. 3. To impart knowledge on networking support for cloud computing, including cloud reference models, data center design, and interconnection networks, with a focus on the integration with the Internet of Things (IoT). 4. To address cloud security challenges comprehensively, covering software-as-a-service security, risk management, security monitoring, and implementing security measures for data, applications, and virtual machines. 5. To explore web-based cloud applications, examining the pros and cons of cloud service development, and utilizing major cloud platforms like Amazon EC2, Google App Engine, and IBM Clouds for practical application.
Course Outcome	1. Students will be able to explain the evolution and system models of cloud computing, describe the NIST cloud computing reference architecture, and identify examples of IaaS, PaaS, and SaaS providers. 2. Students will understand virtualization technologies, including hypervisors such as VMware, KVM, and Xen, and will be capable of implementing virtualization for CPU, memory, I/O devices, servers, desktops, networks, and data centers. 3. Students will grasp the necessary networking support for cloud computing, understand the cloud reference model, and design and interconnect data centers while integrating with IoT. 4. Students will identify and address cloud security challenges, develop security governance and risk management strategies, and implement robust security measures for data, applications, and virtual machines within the cloud environment. 5. Students will gain the ability to develop and deploy web-based cloud applications, evaluate different types of cloud services, and utilize development tools and services from major cloud providers, enhancing their practical skills in cloud service development.

Module Name	Content	No. of Hrs.
Introduction	Evolution of Cloud Computing –System Models for Distributed and Cloud Computing – NIST Cloud Computing Reference Architecture -IaaS – On-demand Provisioning – Elasticity in Cloud – E.g. of IaaS Providers - PaaS – E.g. of PaaS Providers - SaaS – E.g. of SaaS Providers – Public ,Private and Hybrid Clouds..	8
Virtualization Technology	Definition, Understanding and Benefits of Virtualization. Implementation Level of Virtualization, Virtualization Structure/Tools and Mechanisms, Hypervisor, VMware, KVM, Xen. Virtualization of CPU, Memory, I/O Devices, Virtual Cluster and Resources Management, Virtualization of Server, Desktop, Network, and Virtualization of data-center.	10
Networking Support for Cloud Computing	Ubiquitous Cloud and the Internet of Things. Cloud Computing Architecture: Cloud Reference Model, Layer and Types of Clouds, Services models, Data center Design and interconnection Network, Architectural design of Computer and Storage Clouds.	8
Security in the Cloud	Security Overview – Cloud Security Challenges – Software-as-a-Service Security – Security Governance – Risk Management – Security Monitoring – Security Architecture Design – Data Security – Application Security – Virtual Machine Security	8
Web-Based Cloud Application	Web-Based Application, Pros and Cons of Cloud Service Development, Types of Cloud Service Development, Software as a Service, Platform as a Service, Web Services, On-Demand Computing, Discovering Cloud Services, Development Services and Tools, Amazon Ec2, Google App Engine, IBM Clouds	8
	Total No. of Hours	**42**

Textbooks	1. Raj Kumar Buyya, "Cloud Computing: Principles and Paradigms, wiley 2. Barrie Sosinsky, "Cloud Computing Bible", Wiley Publishers
References	1. John W.Rittinghouse and James F.Ransome, "Cloud Computing: Implementation, 2. Michael Miller, "Cloud Computing : Web-based Applications That Change The Way You Work and Collaborate Online", Pearson Education

COMPUTER VISION (SET/CS/BT/E706)		
Course Objective	1. To introduce the fundamental concepts and scope of computer vision, including its historical context and various applications across different fields. 2. To provide a comprehensive understanding of image formation, representation, and basic image processing and analysis techniques, including image acquisition, enhancement, filtering, and segmentation. 3. To educate on feature description and matching techniques, and to delve into object detection and recognition methodologies, enabling students to develop practical skills in these areas. 4. To cover the basics and advanced applications of deep learning in computer vision, focusing on convolutional neural networks (CNNs), training methods, and the use of transfer learning for improving model performance. 5. To explore the ethical and social implications of computer vision technologies, emphasizing privacy concerns, algorithmic bias, and the responsible deployment of these technologies.	
Course Outcome	1. Students will gain a clear understanding of the scope, applications, and historical milestones of computer vision, enabling them to appreciate its significance and evolution. 2. Students will acquire practical skills in image formation, representation, and basic image processing techniques, including the use of cameras and sensors for image acquisition, and methods for enhancing and analyzing digital images. 3. Students will develop the ability to describe and match features using techniques such as SIFT, SURF, and ORB, and will learn to implement object detection methods like Haar cascades and the Viola-Jones framework. 4. Students will be proficient in the fundamentals of deep learning for computer vision, particularly in training convolutional neural networks (CNNs) for tasks such as image classification and object detection, and will understand the application of transfer learning. 5. Students will be equipped to address the ethical and social implications of computer vision technologies, ensuring they consider privacy, fairness, and responsible deployment in their professional practice.	
MODULE	**CONTENT**	**No. of Hrs.**
Introduction to Computer Vision	Definition and scope of computer vision, Overview of applications in various fields, Historical developments and milestones.	8
Image Formation and Representation and Image Processing and Analysis:	Basics of digital images: pixels, resolution, color models, Image acquisition: cameras, sensors, image formats, Image enhancement techniques: filtering, histogram equalization, Image Processing, and Analysis: Image filtering: convolution, Gaussian smoothing, edge detection, Feature extraction: corners, edges, blobs, Image segmentation: thresholding, region-based methods, contour detection.	10
Feature Description and Matching and Object Detection and Recognition	Local feature descriptors: SIFT, SURF, ORB, Feature matching techniques: nearest neighbor, RANSAC, Object Detection: Haar cascades for face detection, Viola-Jones object detection framework.	8
Deep Learning for Computer Vision	Basics of convolutional neural networks (CNNs), Training CNNs for image classification and object detection, Transfer learning and fine-tuning pretrained models.	8
Ethical and Social Implications:	Privacy concerns in computer vision systems, Bias and fairness in algorithmic decision-making, Ethical considerations in deploying computer vision technologies.	6
	Total No. of Hours	40
Text Books	1. David A. Forsyth and Jean Ponce: Computer Vision – A Modern Approach, PHI Learning (Indian Edition), 2009	
References	1. E. R. Davies: Computer and Machine Vision – Theory, Algorithms and Practicalities, Elsevier (Academic Press), 4th edition, 2013. 2. R. C. Gonzalez and R. E. Woods "Digital Image Processing" Addison Wesley 2008. 3. Richard Szeliski "Computer Vision: Algorithms and Applications" Springer-Verlag London Limited 2011.	

Semester VIII

S. No.	**Category**	**Course Code**	**Course Title**	L	T	P	Contact Hrs./Week	Credits
1	Core Subjects	SET/CS/BT/C801	UNIX Shell Programming	3	1		4	4
2			Program Elective-5	3	1	-	4	4
3			Program Elective-6	3	1	-	4	4
4	Life Skills and personality development	SET/CS/BT/L802	Disaster Management	-	-	1	4	2
5	Skill Course	SET/CS/SC/C803	Project and Dissertation			1	12	6
		Total		9	3	2	28	20

S. No.	**Category**	**Course Code**	**Course Title**	L	T	P	Contact Hrs./Week	Credits
1	Program Elective-5	SET/CS/BT/E801	Natural Language Processing	3	1	-	4	4
2		SET/CS/BT/E802	Internet of Things (IoT)	3	1	-	4	4
3		SET/CS/BT/E803	Machine Learning	3	1	-	4	4
4		SET/CS/BT/E804	Big Data Analytics	3	1	-	4	4
1	Program Elective-6	SET/CS/BT/E805	Cyber Security and Ethical Hacking	3	1	-	4	4
2		SET/CS/BT/E806	Mobile Application Development	3	1		4	4
3		SET/CS/BT/E807	Blockchain Technology	3	1		4	4
4		SET/CS/BT/E808	Deep Learning	3	1		4	4

UNIX AND SHELL PROGRAMMING (SET/CSE//BT/C801)		
Course Objective	1. To provide a comprehensive understanding of Unix user accounts, including the process of starting and shutting down processes, logging in and out, and basic command-line operations. 2. To introduce students to shell programming in Unix, covering Unix file systems, file management commands, shell variables, and scripting techniques such as command substitution, functions, conditionals, and loops. 3. To familiarize students with regular expressions and filters, including patterns, syntax, character classes, and quantifiers, and to demonstrate their practical application using tools like egrep, sed, awk, and perl. 4. To equip students with the necessary knowledge and skills to work within the C environment in Unix, including understanding the C compiler, vi editor, project management, memory management, makefiles, static and dynamic libraries, and debugging with gdb. 5. To explore advanced topics related to Unix processes, including process management, initialization processes, job control, network files, security, authentication, password administration, archiving, and handling signals and signal handlers.	
Course Outcome	1. Students will be proficient in Unix user account management, process handling, and basic command-line operations, enabling them to navigate and operate within the Unix environment effectively. 2. Students will have the ability to write shell scripts to automate tasks, manipulate files and directories, and customize their Unix environment using shell variables and scripting constructs. 3. Students will demonstrate competence in using regular expressions and filters to search and manipulate text data efficiently, employing tools like egrep, sed, awk, and perl for text processing tasks. 4. Students will be proficient in working within the C environment in Unix, including compiling and debugging C programs, managing projects with makefiles, and effectively utilizing static and dynamic libraries. 5. Students will have a deep understanding of Unix processes, job control mechanisms, security features, and signal handling, enabling them to develop robust and secure Unix-based applications.	
Module Name	**Content**	**No. of Hrs.**
Unix	User accounts, Unix – starting and shutting processes, Logging in and Logging out, Command line, simple commands.	8
Shell Programming	Unix file system, Unix files, inodes and structure and file system related commands, Shell as command processor, shell variables, creating command substitution, scripts, functions, conditionals, loops, customizing environment.	8
Regular Expressions and Filters	Introducing regular expressions patterns, syntax, character classes, quantifiers, introduction to egrep, sed, programming with awk and perl.	10
The C Environment	The C compiler, vi editor, compiler options, managing projects, memory management, use of make files, dependency calculations, memory management,	10

	dynamic and static memory, building and using static and dynamic libraries, using ldd, soname, dynamic loader, debugging with gdb.	
Processes	Processes, starting and stopping processes, initialization processes, rc and init files, job control – at, batch, cron, time, network files, security, privileges, authentication, password administration, archiving, Signals and signal handlers.	6
	Total No. of Hours	**42**
Text Books	1. Sumitabha Das, "Your Unix – The Ultimate Guide", TMH, 2000.	
References	1. John Goerzen, "Linux Programming Bible", IDG Books, New Delhi, 2000. 2. Mathew, "Professional Linux Programming", Vol.1 & 2, Wrox-Shroff, 2001. 3. Welsh & Kaufmann "Running Linux", O'Reiley& Associates, 2000.	

Program Elective Courses

NATURAL LANGUAGE PROCESSING (SET/CS/BT/E801)		
Course Objective	1. To introduce students to the field of Natural Language Understanding (NLU) and its significance in various applications such as machine translation and database interfaces. 2. To familiarize students with the different levels of language analysis, including syntax, semantics, and pragmatics, and how these levels contribute to language understanding. 3. To provide an understanding of the organization and components of Natural Language Understanding systems, including the linguistic background necessary for building such systems. 4. To explore the fundamentals of semantics and knowledge representation in NLU, including its applications and challenges. 5. To delve into the principles of grammars and parsing in natural language processing, covering different parsing techniques and ambiguity resolution methods.	
Course Outcome	1. Students will gain a comprehensive understanding of the principles and applications of Natural Language Understanding, enabling them to appreciate its importance in various real-world contexts. 2. Students will be able to evaluate and analyze language understanding systems, identifying their strengths and weaknesses based on different evaluation metrics. 3. Students will acquire the necessary knowledge and skills to perform different levels of language analysis, including syntax, semantics, and pragmatics, in the context of NLU. 4. Students will develop proficiency in designing and implementing grammars for natural language processing tasks, as well as utilizing parsing techniques to analyze and understand natural language input. 5. Students will be able to address ambiguity in natural language processing through the application of statistical methods, probabilistic language processing techniques, and semantic analysis, thereby enhancing the accuracy and robustness of NLU systems.	
Module Name	**Content**	**No. of Hrs.**
Introduction to Natural Language Understanding	The study of Language, Applications of NLP, Evaluating Language Understanding Systems, Different levels of Language Analysis, Representations and Understanding, Organization of Natural language Understanding Systems, Linguistic Background: An outline of English syntax	8
Introduction to semantics	Introduction to semantics and knowledge representation, Some applications like machine translation, database interface.	8
Grammars and Parsing	Grammars and sentence Structure, Top-Down and Bottom-Up Parsers, Transition Network Grammars, Top- Down Chart Parsing. Feature Systems and Augmented Grammars: Basic Feature system for English, Morphological Analysis and the Lexicon, Parsing with Features, Augmented Transition Networks	10
Grammars for Natural Language	Auxiliary Verbs and Verb Phrases, Movement Phenomenon in Language, Grammars. Human preferences in Parsing, Encoding uncertainty, Deterministic Parser.	8
Ambiguity Resolution	Statistical Methods, Probabilistic Language Processing, Estimating Probabilities, Part-of-Speech tagging, Obtaining Lexical Probabilities, Probabilistic Context-Free Grammars, Best First Parsing.Semantics and Logical Form, Word senses and Ambiguity, Encoding Ambiguity in Logical Form	8
	Total No of Hours	**42**
Text Books	1. Akshar Bharti, Vineet Chaitanya and Rajeev Sangal, "NLP: A Paninian Perspective", Prentice Hall	
References	1. James Allen, "Natural Language Understanding", Pearson Education 2. D. Jurafsky, J. H. Martin, "Speech and Language Processing", Pearson Education	

INTERNET OF THINGS (IOT) (SET/CS/BT/E802)		
Course Objective	1. Provide an architectural overview of IoT systems, focusing on design principles, capabilities, and standards considerations. 2. Explore the fundamentals of M2M and IoT technologies, including devices, gateways, networking, data management, and analytics. 3. Cover networking fundamentals specific to IoT, including protocols and layers such as PHY/MAC, network, transport, and session layers. 4. Discuss data management and analytics techniques for IoT systems, including data collection, preprocessing, storage, processing, visualization, and machine learning. 5. Examine embedded systems used in IoT devices, programming languages, security threats, vulnerabilities, and techniques to ensure secure IoT systems.	
Course Outcome	1. Understand the architectural principles and capabilities required for designing IoT systems and recognize the importance of standards in IoT development. 2. Identify the components and technologies involved in M2M and IoT systems, including devices, networking protocols, and analytics methods. 3. Demonstrate proficiency in networking fundamentals specific to IoT, including protocols and layers used in IoT communication. 4. Develop skills in data management and analytics for IoT, including collecting, preprocessing, storing, processing, visualizing, and analyzing IoT data. 5. Gain practical knowledge of embedded systems, programming languages, and security measures to design and implement secure IoT solutions.	
MODULE	**CONTENT**	**No. of Hrs.**
IoT-An Architectural Overview:	Building an architecture, Main design principles and needed capabilities, An IoT architecture outline, standards considerations. M2M and IoT Technology Fundamentals- Devices and gateways, Local and wide area networking, Data management, Business processes in IoT, Everything as a Service (XaaS), M2M and IoT Analytics, Knowledge Management.	8
Networking Fundamental of IOT	PHY/MAC Layer (3GPP MTC, IEEE802.11, IEEE 802.15), WirelessHART, ZWave, Bluetooth Low Energy, Zigbee Smart Energy, DASH7 - Network Layer-IPv4, IPv6, 6LoWPAN, 6TiSCH, ND, DHCP, ICMP, RPL, CORPL, CARP, Transport and Session Layer: Transport Layer (TCP, MPTCP, UDP, DCCP, SCTP)- (TLS, DTLS) – Session Layer-HTTP, CoAP, XMPP, AMQP, MQTT.	12
Data Management and Analytics	Data collection, storage, and preprocessing, Real-time and batch processing, Data visualization techniques, Introduction to analytics and machine learning for IoT data.	6
Embedded Systems and Security of IOT	Introduction to microcontrollers (Arduino, Raspberry Pi, etc.), Sensors and actuators used in IoT devices, Programming embedded systems (C/C++, Python, etc.), Security: Threats and vulnerabilities in IoT systems, Authentication and access control, Encryption techniques, Privacy considerations and regulations.	8
Cloud Computing and Edge Computing and IOT Platform	Introduction to cloud platforms (AWS, Azure, Google Cloud, etc.), Edge computing concepts and architectures, Deployment strategies for IoT applications, IOT Platform: Overview of IoT platforms (IoTivity, ThingWorx, IBM Watson IoT, etc.).	8
Total No. of Hours		42
Text Books	1. Hakima Chaouchi, — "The Internet of Things Connecting Objects to the Web" ISBN : 978-1-84821-140-7, Wiley Publications 2. Olivier Hersent, David Boswarthick, and Omar Elloumi, — "The Internet of Things: Key Applications and Protocols", WileyPublications 3. Vijay Madisetti and ArshdeepBahga, — "Internet of Things (A Hands-on-Approach)",1 st Edition, VPT, 2014. 4. J. Biron and J. Follett, "Foundational Elements of an IoT Solution", O'Reilly Media,2016. Keysight Technologies, "The Internet of Things: Enabling Technologies and Solutions for Design and Test", Application Note, 2016.	
References	1. Daniel Minoli, — "Building the Internet of Things with IPv6 and MIPv6: The Evolving World of M2M Communications", ISBN: 978-1-118-47347-4, Willy Publications 2. Pethuru Raj and Anupama C. Raman, "The Internet of Things: Enabling Technologies, Platforms,	

MACHINE LEARNING (SET/CS/BT/E803)		
Course Objective	1. Define learning systems and explore their goals and applications within the context of machine learning. 2. Examine the various aspects involved in developing a learning system, including training data acquisition, concept representation, and function approximation techniques. 3. Introduce logistic regression and other discriminative learning algorithms, including perceptron, decision trees, and support vector machines. 4. Discuss unsupervised learning techniques such as clustering, dimensionality reduction, and latent variable models. 5. Explore reinforcement learning and control methods, including Markov decision processes (MDPs), Q-learning, and policy iteration.	
Course Outcome	1. Understand the fundamental concepts and objectives of learning systems and their relevance in machine learning applications. 2. Gain proficiency in acquiring and preprocessing training data, representing concepts, and approximating functions for learning tasks. 3. Identify and implement various discriminative and generative learning algorithms for classification and regression tasks. 4. Apply unsupervised learning techniques to uncover patterns, clusters, and latent variables in data sets. 5. Develop a comprehensive understanding of reinforcement learning algorithms and their applications in decision-making and control tasks.	
Module Name	**Content**	**No. of Hrs.**
Introduction	Definition of learning systems, Goals and applications of machine learning. Aspects of developing a learning system: training data, concept representation, function approximation. Definition of learning systems, Goals and applications of machine learning. Aspects of developing a learning system: training data, concept representation, function approximation. linear discriminative, non-linear discriminative, decision trees, probabilistic (conditional and generative), nearest neighbor	10
Logistic regression	Logistic regression, Perceptron, Exponential family, Generative learning algorithms, Gaussian discriminant analysis, Naive Bayes, Support vector machines: Optimal hyper plane, Kernels. Model selection and feature selection. Combining classifiers: Bagging, boosting (The Ada boost algorithm), Evaluating and debugging learning algorithms, Classification errors	12
Unsupervised learning:	Clustering. K-means. EM Algorithm. Mixture of Gaussians. Factor analysis. PCA (Principal components analysis), ICA (Independent components analysis), latent semantic indexing. Spectral clustering, Markov models Hidden Markov models (HMMs).	10
Reinforcement Learning and Control	MDPs. Bellman equations, Value iteration and policy iteration, Linear quadratic regulation (LQR). LQG. Q-learning. Value function approximation, Policy search. Reinforce. POMDPs	10
	Total No. of Hours	**42**
Text books	1. Tom M Mitchell, Machine Learning, McGraw Hill Education. 2. Duda, Richard, Pattern Classification. 2nd,Wiley India	
References	1. Tom M. Mitchell, Machine Learning . McGraw-Hill Series,. 2. Introduction to Machine Learning – Ethem Alpaydin, MIT Press, Prentice hall of India.	

BIG DATA ANALYTICS (SET/CS/BT/E804)

Course Objective	1. Provide an understanding of digital data types and the challenges of managing big data. 2. Introduce the concepts and tools of big data analytics, including Apache Hadoop and IBM BigInsights. 3. Explore the design and architecture of Hadoop Distributed File System (HDFS) and its interfaces. 4. Familiarize students with MapReduce programming paradigm and its implementation in Hadoop. 5. Introduce other big data processing tools like Pig, Hive, HBase, and Big SQL, along with their features and use cases.
Course Coutcome	1. Understand the fundamentals of big data analytics, including the history of Hadoop and its ecosystem. 2. Gain proficiency in managing and analyzing data using Unix tools and Hadoop. 3. Develop skills in designing, implementing, and optimizing MapReduce jobs for data processing. 4. Learn to use Pig, Hive, HBase, and Big SQL for different data processing tasks and scenarios. 5. Explore machine learning techniques in big data analytics and understand their applications in supervised and unsupervised learning, as well as collaborative filtering.

Module Name	Content	No. of Hrs.
INTRODUCTION TO BIG DATA AND HADOOP	Types of Digital Data, Introduction to Big Data, Big Data Analytics, History of Hadoop, Apache Hadoop, Analysing Data with Unix tools, Analysing Data with Hadoop, Hadoop Streaming, Hadoop Echo System, IBM Big Data Strategy, Introduction to Infosphere Big Insights and Big Sheets	8
HDFS(Hadoop Distributed File System)	The Design of HDFS, HDFS Concepts, Command Line Interface, Hadoop file system interfaces, Data flow, Data Ingest with Flume and Scoop and Hadoop archives, Hadoop I/O: Compression, Serialization, Avro and File-Based Data structures	10
Map Reduce	Anatomy of a Map Reduce Job Run, Failures, Job Scheduling, Shuffle and Sort, Task Execution, Map Reduce Types 0and Formats, Map Reduce Features	8
Hadoop Eco System	Pig : Introduction to PIG, Execution Modes of Pig, Comparison of Pig with Databases, Grunt, Pig Latin, User Defined Functions, Data Processing operators. Hive : Hive Shell, Hive Services, Hive Metastore, Comparison with Traditional Databases, HiveQL, Tables, Querying Data and User Defined Functions. Hbase :HBasics, Concepts, Clients, Example, Hbase Versus RDBMS. Big SQL : Introduction	8
Data Analytics with R	Machine Learning : Introduction, Supervised Learning, Unsupervised Learning, Collaborative Filtering.Big Data Analytics with BigR.	8
	Total No. of Hours	**42**

Textbooks	1. Big data,BlackBook:Covers Hadoop 2,map reduce,Hive,YARN,PIG,R and data Visualization, Dreamtech, Wiley India 2. Seema Acharya, "Big Data Analytics" Wiley 2015.
References	3. Tom Plunkett, Mark Hornick, "Using R to Unlock the Value of Big Data: Big Data Analytics with Oracle REnterprise and Oracle R Connector for Hadoop", McGraw-Hill/Osborne Media (2013), Oracle press.

CYBER SECURITY AND ETHICAL HACKING (SET/CS/BT/E805)		
Course Objective	1. Provide an introduction to hacking, including its types and processes, and the basics of security. 2. Establish a foundation for ethical hacking techniques, covering methodologies and various attack vectors. 3. Explore web application security, including core defense mechanisms and techniques for managing and securing web applications. 4. Examine hacking techniques specific to wireless networks, including sniffing, spoofing, and denial of service attacks. 5. Discuss the applications of hacking techniques in various scenarios, such as firewall engineering, secure communications, and mobile hacking.	
Course Outcome	1. Understand the fundamentals of hacking, including its various types and the importance of security. 2. Gain proficiency in ethical hacking methodologies, including social engineering, password hacking, and network penetration testing. 3. Develop skills in identifying and mitigating web application security vulnerabilities. 4. Acquire knowledge of hacking techniques specific to wireless networks and best practices for securing wireless LANs. 5. Learn about the applications of hacking techniques in real-world scenarios and the legal and ethical considerations involved.	
Module Name	**Content**	**No. of Hrs.**
BASICS OF HACKING	Introduction to Hacking, Types of Hacking, Hacking Process, Security – Basics of Security- Elements of Security, Penetration Testing, Scanning, ExploitationWebBased Exploitation.	6
ETHICAL HACKING TECHNIQUES	Building the foundation for Ethical Hacking, Hacking Methodology, Social Engineering, Physical Security, Hacking Windows, Password Hacking, Privacy Attacks, Hacking the Network, Hacking Operating Systems- Windows & Linux, Application Hacking, Footprinting, Scanning, Enumeration.	9
WEB APPLICATIONS SECURITY	Evolution of Web applications, Web application security, Core DefenseMechanisms, Managing the Application, Web Application Technologies- Web Hacking, Web functionality, How to block content on the Internet, Web pages through Email, Web Messengers, Unblocking applications, Injecting CodeInjecting into SQL, Attacking Application Logic.	9
HACKING TECHNIQUES IN WIRELESS NETWORKS	Introduction to Wireless LAN Overview, Wireless Network Sniffing, WirelessSpoofing, Port Scanning, Wireless Network Probing, AP Weakness, Denial of Service (DOS), Man-in-the-Middle Attacks, War Driving, Wireless Security Best Practices, Software Tools, Cracking WEP, Cracking WPA & WPA-II.	9
HACKING TECHNIQUES APPLICATIONS	Safer tools and services, Firewalls, Filtering services, Firewall engineering, Secure communications over insecure networks, Case Study: Mobile HackingBluetooth-3Gnetwork weaknesses, Case study: DNS Poisoning, Hacking Laws	9
	Total No. of Hours	**42**

Textbooks	1. Kevin Beaver, "Hacking for Dummies" Second Edition, Wiley Publishing, 2. Stuart McClure, Joel Scambray, George Kurtz, "Hacking Exposed 6: NetworkSecurity Secrets & Solutions", Seventh edition, McGraw-Hill Publisher
References	1. Ankit Fadia, "An Unofficial Guide to Ethical Hacking" Second Edition, Macmillan publishers India Ltd, 2006 2. Ankit Fadia, "How to Unblock Everything on the Internet" Vikas PublishingHouse Pvt Ltd, 2012

MOBILE APPLICATION DEVELOPMENT(SET/CS/BT/E806)

Course Objective	1. Introduce students to mobile application development, focusing on Java technology for rich client applications. 2. Provide an in-depth understanding of the Android platform, its features, architecture, and various versions. 3. Familiarize students with Android Studio, the development environment, and essential tools like the Android Debug Bridge (ADB). 4. Cover the basics of Android application framework, including resource layout, manifest file, and activity life-cycle. 5. Enable students to design user interfaces for Android applications using different layout components and Material Design principles.
Course Outcome	1. Gain proficiency in developing Android applications using Java technology, starting from the basics to advanced concepts. 2. Understand the architecture and features of the Android platform and its evolution through different versions. 3. Acquire practical skills in using Android Studio for application development and setting up emulators. 4. Learn various techniques for data persistence in Android applications, including shared preferences, file handling, and SQLite database management. 5. Develop the ability to implement background processes, networking, telephony services, and multitasking in Android applications using services, threads, and broadcast receivers.

Module Name	Content	No. of Hrs.
JAVA TECHNOLOGY FOR RICH CLIENT APPLICATIONS	Introduction to mobile application development, trends, introduction to various platforms, introduction to smart phones. Android platform features and architecture, versions, comparison added features in each version. ART (Android Runtime), ADB (Android Debug Bridge). Android studio and its working environment, grade build system, emulator setup. Application framework basics: resources layout, values, asset XML representation and generated R.Javafile, Android manifestfile. Creating a simple application	10
ANDROID User Interface Design	GUI for Android: Introduction to activities, activities life-cycle, and Android v7 support library form API21 for lower version support. Intent object, intent filters, adding categories, linking activities, user interface design components Views and View Groups: Basic views, picker views, adapter views, Menu, AppBar etc., basics of screen design; different layouts. App widgets. Lollipop Material design: new themes, new widgets, Card layouts. Recycler View ,Fragments: Introduction to activities, activities life-cycle.	12
DATA PERSISTENCE	Different Data persistence schemes: Shared preferences, File Handling, Managing data using SQLite database Content providers: user content provider, Android in build content providers.	8
BACK GROUND RUNNING PROCESS, NETWORKING AND TELEPHONY SERVICES	Services: introduction to services – local service, remote service and binding the service, the communication between service and activity, Intent Service Multithreading: Handlers ,Async Task, Android network programming :Http Url Connection, Connecting to REST based and SOAP based Web services Broad cast receivers: Local Broadcast Manager, Dynamic broadcast receiver, System Broadcast. Pending Intent, Notifications, Telephony Manager: Sending SMS and making calls.	12
	Total No. of Hours	**42**

Textbooks	1. Lee," Beginning android 4 application development "ISBN 9788126535576 Wiley India 2. Greg Milette, Adam Stroud, "PROFESSIONAL Android™ Sensor Programming", John Wiley
Reference Books	1. Paul Deital, Harvey Deital, Alexander Wald, "Android 6 for Programmers ,App Driven approach", 2015, Prentice Hall 2. Dutson"Android Development Patterns: Best Practices for Professional Developers" Pearson

BLOCKCHAIN TECHNOLOGY (SET/CS/BT/E807)		
Course Objective	1. To provide an understanding of blockchain technology, including its history, characteristics, and types, as well as the basics of cryptography. 2. To explore the architecture of blockchain systems, including their components, decentralization, consensus mechanisms, and scalability challenges. 3. To introduce Bitcoin and other cryptocurrencies, covering their underlying technologies, transactions, mining, wallets, and differences. 4. To delve into Ethereum and smart contracts, including the Ethereum platform, decentralized applications (DApps), Solidity programming language, and testing smart contracts. 5. To familiarize students with Hyperledger Fabric development, including its architecture, chaincode development, membership services, and interaction with Fabric networks.	
Course Outcome	1. Students will gain a comprehensive understanding of blockchain technology and cryptography basics, enabling them to appreciate the principles underlying decentralized systems and secure transactions. 2. Students will be able to analyze blockchain architectures, identify their components, and evaluate different consensus mechanisms and scalability solutions. 3. Students will understand the fundamentals of Bitcoin and other cryptocurrencies, capable of explaining their functionalities and distinguishing characteristics. 4. Students will acquire practical skills in Ethereum development, including setting up a development environment, writing smart contracts in Solidity, and interacting with the Ethereum network. **5.** Students will be proficient in Hyperledger Fabric development, able to design, deploy, and interact with Fabric networks, including writing chaincode and managing membership services.	
MODULE	**CONTENT**	**No. of Hrs.**
Introduction to Blockchain and Cryptography Basics	What is Blockchain, History and evolution of blockchain technology, Characteristics of blockchain, Types of blockchains: public, private, and consortium, Use cases and applications of blockchain technology, Cryptography: Introduction to cryptography, Hash functions and cryptographic hashing, Public key cryptography, Digital signatures and certificates, Merkle trees and their role in blockchain.	8
Blockchain Architecture	Components of a blockchain: blocks, transactions, nodes, Decentralization and distributed consensus, Understanding blockchain networks, Blockchain protocols: Proof of Work (PoW), Proof of Stake (PoS), etc., Blockchain scalability challenges and solutions.	6
Bitcoin and Cryptocurrencies	Overview of Bitcoin and its underlying technology, Bitcoin transactions and mining, Bitcoin wallets and addresses, Other cryptocurrencies and their differences from Bitcoin.	6
Ethereum and Smart Contracts	Introduction to Ethereum platform, Ethereum Virtual Machine (EVM), Decentralized applications (DApps), Setting up Ethereum development environment, Introduction to Truffle framework, Smart Contracts: Smart contracts: definition, features, and applications, Solidity programming language, interacting with smart contracts using web3.js, Testing smart contracts.	10
Hyperledger Fabric Development	Introduction to Hyperledger Fabric, Fabric architecture: channels, peers, ordering service, Chaincode (smart contract) development in Fabric, Membership services and identity management, setting up Hyperledger Fabric network, Writing, chaincode in Go or Node.js, Interacting with Fabric network using SDKs.	10
	Total No. of Hours	40

Text Books	1. Mark Gates, "Block chain: Ultimate guide to understanding block chain, bit coin, crypto currencies, smart contracts and the future of money", Wise Fox Publishing and Mark Gates 2017. 2. Salman Baset, Luc Desrosiers, Nitin Gaur, Petr Novotny, Anthony O'Dowd, Venkatraman Ramakrishna, "Hands-On Block chain with Hyper ledger: Building decentralized applications with Hyperledger Fabric and Composer", 2018. 3. Bahga, Vijay Madisetti, "Block chain Applications: A Hands-On Approach", Arshdeep Bahga, Vijay Madisetti publishers 2017.
References	1. Andreas Antonopoulos, "Mastering Bitcoin: Unlocking Digital Crypto currencies", O'Reilly Media, Inc. 2014. 2. Melanie Swa, "Block chain ",O'Reilly Media 2014

DEEP LEARNING (SET/CS/BT/E808)		
Course Objective	1. To provide an overview of deep learning, including its historical context, evolution, and applications across various domains. 2. To cover the fundamentals of neural networks, including feedforward networks, activation functions, loss functions, and optimization algorithms. 3. To introduce convolutional neural networks (CNNs) and recurrent neural networks (RNNs), along with their architectures, training techniques, and applications. 4. To explore generative adversarial networks (GANs) and reinforcement learning (RL), including their architectures, training methods, and real-world applications. 5. To delve into autoencoders and variational autoencoders (VAEs), discussing their principles, variants, and applications in dimensionality reduction and anomaly detection.	
Course Outcome	1. Students will gain a comprehensive understanding of deep learning principles, frameworks, and applications, enabling them to appreciate its significance in solving real-world problems. 2. Students will be proficient in designing and training neural networks, understanding the importance of activation functions, loss functions, and optimization algorithms in model performance. 3. Students will have practical experience with convolutional neural networks and recurrent neural networks, capable of implementing and optimizing these architectures for image analysis, sequence modeling, and natural language processing tasks. 4. Students will be familiar with generative adversarial networks and reinforcement learning techniques, equipped to apply these methods in various domains such as image generation, style transfer, and game playing. 5. Students will understand the concepts of auto encoders and variation auto encoders, able to apply them for tasks like dimensionality reduction and anomaly detection, thereby enhancing their skills in unsupervised learning techniques.	
MODULE	**CONTENT**	**No. of Hrs.**
Introduction to Deep Learning	Overview of artificial neural networks (ANNs), Historical context and evolution of deep learning, Applications of deep learning in various fields, Introduction to deep learning frameworks (Tensor Flow, PyTorch, Keras, etc.)	8
Fundamentals of Neural Networks and Training Deep Neural Networks	Basics of feed forward neural networks, Activation functions and their properties, Loss functions for regression and classification tasks, Optimization algorithms (Gradient Descent, Adam, RMSProp, etc.) Deep Neural Network: Back propagation algorithm, Vanishing and exploding gradients problem, Weight initialization techniques, Batch normalization and regularization methods.	10
Introduction to Convolutional Neural Networks (CNNs) and Recurrent	Introduction to CNNs, Convolutional layers, pooling layers, and activation functions, Architecture of popular CNN models (LeNet, AlexNet, VGG, ResNet, etc.), Transfer learning with pre-trained CNNs, Introduction to RNNs, Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU), Applications of RNNs in	10

Neural Networks (RNNs)	sequence modeling, time series analysis, and natural language processing, Training and optimizing RNNs.	
Generative Adversarial Networks (GANs) and Introduction to Reinforcement Learning	Introduction to GANs, Architecture of GANs (Generator, Discriminator), Training GANs and common challenges (mode collapse, instability), Applications of GANs in image generation, style transfer, etc, RNN: Introduction to reinforcement learning (RL), Markov Decision Processes (MDPs) and the RL framework, Q-learning, Policy Gradient methods, Deep Q-Networks (DQN) and Deep Deterministic Policy Gradient (DDPG).	10
Autoencoders and Variational Autoencoders (VAEs)	Introduction to autoencoders, Variants of autoencoders (Sparse autoencoders, Denoising autoencoders), Introduction to VAEs and generative modelling, Applications of autoencoders and VAEs in dimensionality reduction, anomaly detection, etc.	8
Total No. of Hours		46
Text Book	1. Ian Goodfellow and Yoshua Bengio and Aaron Courville. Deep Learning. An MIT Press book. 2016. 2. Charu C. Aggarwal. Neural Networks and Deep Learning: A Textbook. Springer. 2019.	
Reference	1. Ian Goodfellow and Yoshua Bengio and Aaron Courville. Deep Learning. An MIT Press book. 2016. 2. Charu C. Aggarwal. Neural Networks and Deep Learning: A Textbook. Springer. 2019.	