Sharnbasva University, Kalaburagi
Scheme of Teaching and Examination 2018-19
Outcome Based Education(OBE) and Choice Based Credit System (CBCS) (Effective from the academic year 2018-19)

III SEMESTER B. Tech.
(Common to Computer Science & Engineering and Information Science & Engineering)

Sl. No.	**Course Code**		**Course Title**	**Teaching Dept. & Paper Setting Board**	**Teaching Hours/week**			**Examination**				**Credits**
					L	T	P	**Duratio n in hours**	**CIE Marks**	**SEE Marks**	**Total Marks**	
1.	BSC	18MAT31	Engineering Mathematics-III	Mathematics	3	2		3	50	50	100	04
2.	PCC	18CS32	Data structures in C and Applications	CSE/ISE	3	2		3	50	50	100	04
3.	PCC	18CS33	Electronics Circuits and Logic Design	ECE/CSE /ISE	3	2		3	50	50	100	04
4.	PCC	18CS34	Computer Organization and Architecture	CSE/ISE	3	2		3	50	50	100	04
5.	PCC	18CSL35	Data Structures Lab	CSE/ISE			2	3	50	50	100	01
6.	PCC	18CSL36	Electronics Circuits and Logic Design Lab	CSE/ISE			2	3	50	50	100	01
7.	PCC	18CSL37	UNIX Shell Programming Lab	CSE/ISE			2	3	50	50	100	01
8.	PRJ	18CSP38	Project – III	CSE/ISE			2	3	50	50	100	01
9.	HSMC	18KANKK310 / 20KANAK310	Aayda Kathegalu Kannada Kali - 3	Humanities			2	2	50	50	100	01
Total					**12**	**8**	**10**	**26**	**450**	**450**	**900**	**21**

BSC-Basic Science, PCC-Professional Core, HSMC-Humanity and Social Science, PR-Project

Sharnbasva University, Kalaburagi
Scheme of Teaching and Examination 2018-19
Outcome Based Education(OBE) and Choice Based Credit System (CBCS)
(Effective from the academic year 2018-19)

IV SEMESTER B. Tech.
(Common to Computer Science & Engineering and Information Science & Engineering)

Sl. No.	Course Code		Course Title	Teaching Dept. & Paper Seting Board	Teaching Hours/week			Examination				Credits
					L	T	P	Duration in hours	CIE Marks	SEE Marks	Total Marks	
1.	BSC	18MAT41	Engineering Mathematics-IV	Mathematics	3	2		3	50	50	100	04
2.	PCC	18CS42	Design and Analysis of Algorithms	CSE/ISE	3	2		3	50	50	100	04
3.	PCC	18CS43	Microprocessor	CSE/ISE	3	2		3	50	50	100	04
4.	PCC	18CS44	Java Programming	CSE/ISE	3	2		3	50	50	100	04
5.	PCC	18CSL45	Microprocessor Lab	CSE/ISE			2	3	50	50	100	01
6.	PCC	18CSL46	Java Programming lab	CSE/ISE			2	3	50	50	100	01
7.	PCC	18CSL47	Algorithm analysis and design Lab	CSE/ISE			2	3	50	50	100	01
8.	PRJ	18CSP48	Project-IV	CSE/ISE			2	3	50	50	100	01
9.	HSMC	18KANKK410 / 20KANAK410	Mahadasohi Kannada Kali - 4	Humanities			2	2	50	50	100	01
Total					**12**	**8**	**10**	**26**	**450**	**450**	**900**	**21**

BSC-Basic Science, PCC-Professional Core, HSMC-Humanity and Social Science, MP-Mini project

Sharnbasva University, Kalaburagi
Scheme of Teaching and Examination 2018-19
Outcome Based Education(OBE) and Choice Based Credit System (CBCS) (Effective from the academic year 2018-19)

V SEMESTER B. Tech.

(Common to Computer Science & Engineering and Information Science & Engineering)

Sl. No	Course Code		Course Title	Teaching Dept. & Paper Setting Board	Teaching Hours/ week			Examination				Credits
					L	T	P	Duration in hours	CIE Marks	SEE Marks	Total Marks	
1.	PCC	18CS51	Database Management System	CSE/ISE	3	2		3	50	50	100	04
2.	PCC	18CS52	Computer Networks	CSE/ISE	3	2		3	50	50	100	04
3.	PEC	18CS53X	Professional Elective – I	CSE/ISE	3			3	50	50	100	03
4.	OEC	18CS54X	Open Elective – I	CSE/ISE	3			3	50	50	100	03
5.	PCC	18CS55	Web Programming Lab	CSE/ISE			2	3	50	50	100	01
6.	PCC	18CSL56	Database management system lab	CSE/ISE			2	3	50	50	100	01
7.	PEC	18CSL57	Computer Networks lab	CSE/ISE			2	3	50	50	100	01
8.	PRJ	18CSP58	Project-V	CSE/ISE			2	3	50	50	100	01
9.	HSMC	18HSM59	Soft Skills	Humanities			4	2	50	50	100	01
Total					**12**	**4**	**12**	**26**	**450**	**450**	**900**	**19**
	PCC-Professional Core, PEC- Professional Elective, OEC- Open Elective, HSMC-Humanity and Social Science, PR-Project											

Professional Elective – I			**Open Elective – I**		
Sl. No.	**Sub. Code**	**Sub. Name**	**Sl. No.**	**Sub. Code**	**Sub. Name**
1.	18CS531	Automata Theory and Computability	1.	18CS541	Discrete Mathematical Structures and Graph Theory
2.	18CS532	Cloud Computing	2.	18CS542	Microcontroller and Embedded Systems
3.	18CS533	UNIX System Programming	3.	18CS543	Mobile Application Development
4.	18CS534	Social Network Analysis	4.	18CS544	Green Technology

Sharnbasva University, Kalaburagi

Scheme of Teaching and Examination 2018-19

Outcome Based Education(OBE) and Choice Based Credit System (CBCS) (Effective from the academic year 2018-19)

VI SEMESTER B. Tech. (Computer Science & Engineering)

Sl.No	Course Code		Course Title	Teaching Dept. & Paper Setting Board	Teaching Hours/week			Examination				Credits
					L	T	P	Duration in hours	CIE Marks	SEE Marks	Total Marks	
1.	PCC	18CS61	System Software and Complier Design	CSE/ISE	3	2		3	50	50	100	04
2.	PEC	18CS62X	Professional Elective-II	CSE/ISE	3			3	50	50	100	03
3.	PEC	18CS63X	Professional Elective-III	CSE/ISE	3			3	50	50	100	03
4.	OEC	18XX64X	Open Elective –II	CSE/ISE	3			3	50	50	100	03
5.	PCC	18CSL65	System software and Compiler Design Lab	CSE/ISE			2	3	50	50	100	01
6.	PEC	18CSL66	Operating System and UNIX system Programming Lab	CSE/ISE			2	3	50	50	100	01
7.	PEC	18CSL67	Python Lab	CSE/ISE			2	3	50	50	100	01
8.	PRJ	18CSP68	Project-6	CSE/ISE			2	3	50	50	100	01
9.	HSMC	18HSM69	Professional Ethics	Humanities			2	2	50	50	100	01
Total					**12**	**2**	**10**	**26**	**450**	**450**	**900**	**18**
	PCC-Professional Core, PEC- Professional Elective, OEC- Open Elective, HSMC-Humanity and Social Science, PR-Project											

Professional Elective – II			Professional Elective - III		Open Elective – II	
Sl. No.	**Sub. Code**	**Sub. Name**	**Sub. Code**	**Sub. Name**	**Sub. Code**	**Sub. Name**
1.	18CS621	Operating System	18CS631	Rapid programming application using Python	18CS641	Software Engineering
2.	18CS622	Software Testing	18CS632	Sensors and Application	18CS642	Multi Core Architecture
3.	18CS623	Cryptography and Network Security	18CS633	Computer Vision	18CS643	Network Programming

4.	18CS624	Computer Graphics and Visualization	18CS634	Probability, Statistics and Queuing Theory	18CS644	Mobile Computing

Sharnbasva University, Kalaburagi
Scheme of Teaching and Examination 2018-19
Outcome Based Education (OBE) and Choice Based Credit System (CBCS)
(Effective from the academic year 2018-19)

VII SEMESTER B. Tech. (Computer Science & Engineering)

Sl. No.	**Course Code**		**Course Title**	**Teaching Dept. & Paper Setting Board**	**Teaching Hours/ Week**			**Examination**				**Credits**
					L	T	P	**Duration in hours**	**CIE Marks**	**SEE Marks**	**Total Marks**	
1.	PCC	18CS71	Artificial Intelligence & Machine Learning	CSE/ISE	3	2		3	50	50	100	04
2.	PEC	18CS72	Big Data Analytics	CSE/ISE	3	2		3	50	50	100	04
3.	PEC	18CS73X	Professional elective –IV	CSE/ISE	3			3	50	50	100	03
4.	OEC	18XX74X	Open elective –III	CSE/ISE	3			3	50	50	100	03
5.	PCC	18CSL75	Artificial Intelligence & Machine Learning Lab	CSE/ISE			2	3	50	50	100	01
6.	PEC	18CSL76	Big Data Analytics Lab	CSE/ISE			2	3	50	50	100	01
7.	PEC	18CSL77	AWS Cloud Lab	CSE/ISE			2	3	50	50	100	01
8.	PRJ	18CSP78	Project – 7	CSE/ISE			2	3	50	50	100	01
9.	HSMC	18HSM79	Industry Psychology and organizational Behavior	Humanities			2	2	50	50	100	01
Total					**12**	**4**	**10**	**26**	**450**	**450**	**900**	**19**

Note:- Project 7-Real life problem solving project / Research Project/ Field Project

PCC-Professional Core, PEC- Professional Elective, OEC- Open Elective, HSMC-Humanity and Social Science

Professional Elective - V		Open Elective – II	
Sub. Code	Sub. Name	Sub. Code	Sub. Name
18CS731	Data Mining and Data Warehousing	18CS741	Internet of Things
18CS732	AWS Cloud	18CS742	Blockchain Technology
18CS733	System Modelling and Simulation	18CS743	Python Application Programming
18CS734	Storage Area Network	18CS744	Neural Networks and Deep Learning

Sharnbasva University, Kalaburagi
Scheme of Teaching and Examination 2018-19
Outcome Based Education (OBE) and Choice Based Credit System (CBCS)
(Effective from the academic year 2018-19)

VIII SEMESTER B. Tech. (Computer Science & Engineering)

Sl.No	Course Code		Course Title	Teaching Dept. & Paper Setting Board	Teaching Hours/ Week			Examination				Credits
					Training/ Learning/ Practice/ Implementation			Duration in hours	CIE Marks	SEE Marks	Total Marks	
1.	Project	18CSP81	Project Work	4	-		2	3	50	50	100	08
2.	Internship	18CSI82	Internship	12	-		-	3	50	50	100	13
Total				**16**	**-**		**4**	**6**	**100**	**100**	**200**	**21**

Note: - Project 8-Manufacturable and marketable project/ Research Project/ Field Project

SHARNBASVA UNIVERSITY

REGULATIONS FOR CHOICE BASED CREDIT SYSTEM (CBCS) FOR THE UNDERGRADUATE PROGRAMS, 2018-19 (B.TECH.)

1. DEFINITIONS OF KEY WORDS:

i. "**University"** means the Sharnbasva University, Kalaburagi;

ii. "**Academic Year"** is divided into three semesters viz; Two main semesters (Odd and Even Semesters) and One supplementary semester (also called as summer semester).

iii. "**Semester"** Duration of each main semester will be of 19 weeks and that of a supplementary semester will be of 8 weeks. The activities in each semester shall include: (a) Registration of courses in the first week of semester, dropping the courses in the middle and withdrawal from courses towards the end by the students, under the advice of faculty, (b) Teaching, learning, examination and evaluation.

iv. "**Choice Based Credit System" (CBCS)** means Choice Based Credit System which provides choice for the students to select from prescribed courses (core, electives and foundation courses).

v. "**Credit Based Semester System" (CBSS)** Under the CBSS, the requirement for awarding a degree or certificate is prescribed in terms of number of credits to be earned by the students.

vi. "**Course or Subject"** Usually referred to, as 'papers' is a component of a programme. All courses need not carry the same weightage. The courses should define learning objectives and learning outcomes. A course may be designed to comprise lectures / tutorials / laboratory work / field work / outreach activities / project work / internship training / viva / seminars / term papers / assignments / presentations / self-study etc. or a combination of some of these including online courses.

vii. "**CIE"** and **"SEE"** means respectively the Continuous Internal Evaluation and Semester End Examination of the University.

viii. "**First Attempt"** referred to a student who has completed all formalities and passed all the heads in SEE in single attempt, shall be considered as first attempt.

ix. "**Convocation"** means the convocation of the University, where the Degrees, Honorary Degrees, Diplomas, Academic Distinctions and Certificates are awarded as per the requirements of the University.

x. "**Letter Grade"** means an index of the performance of students in a said course. Grades are denoted by letters O, S, A, B, C, D, E and F.

xi. "**Grade Point"** means a numerical weight allotted to each letter grade on 10-point scale.

xii. "**Credit"** means a unit by which the course is measured. It determines the number of hours of instructions required per week. One credit is equivalent to one hour of teaching (lecture)/tutorial or two hours of practical work/field work etc.,per week.

xiii. "**Credit Point"** means the product of grade point and number of credits for a course.

xiv. "**Semester Grade Point Average" (SGPA):** It is a measure of academic performance of student/s in a semester. It is the ratio of total credit points secured by a student in various courses registered in a semester and the total course credits taken during that semester. It shall be expressed up to two decimal places.

xv. "**Cumulative Grade Point Average" (CGPA):** It is a measure of overall cumulative performance of a student over all semesters. The CGPA is the ratio of total credit points earned by a student in various courses in all semesters and the sum of the total credits of all courses in all the semesters. It is expressed up to two decimal places.

xvi. "**Programme"** means an educational programme leading to award of a degree or certificate or diploma.

xvii. "**Transcript or Grade Card or Certificate":** Based on the grades earned, a grade certificate shall be issued to all the registered students after every semester. The grade certificate will display the course details (code, title, number of credits, grade secured) along with SGPA of that semester and CGPA earned till that semester.

xviii. "**Notification"** means the notification of the University.

xix. "**Degree"** means a degree awarded by the University with or without specialization and/ or Minor.

xx. "**Students"** means a person admitted to and pursuing a specified programme of study in the University.

xxi. "**Teacher", "Course Instructor"** means respectively a faculty appointed for imparting instruction and research guidance to students in the University and the Teacher instructing a Course.

xxii. "**OBE**", means **Outcome Based Education.**

xxiii. **"AICTE"** means the All India Council for Technical Education.

xxiv. **"MHRD"** means the Ministry of Human Resource and Development.

xxv. **"Government"** means the Government of Karnataka.

xxvi. **"UGC"** means the University Grants Commission established at New Delhi by an Act of Parliament in 1956.

2. ACADEMIC YEAR:

i. The academic year is divided into three semesters viz; Two main semesters (Odd and Even Semesters) and One supplementary semester. Duration of each main semester will be of 19 weeks and that of a supplementary semester will be of 8 weeks.

ii. The activities in each semester shall include: (a) Registration of courses in the first week of semester, dropping the courses in the middle and withdrawal from courses towards the end by the students, under the advice of faculty, (b) Teaching, learning, examination and evaluation.

3. SEMESTER SYSTEM AND CHOICE BASED CREDIT SYSTEM:

i. Semester wise credit based system shall be followed in each program of study except in the case of certificate and non-degree programs.

ii. Every course offered shall have four components associated with the teaching-learning process, viz; Lecture-L ,Tutorial -T, Laboratory –P, Self-study-S/Assignments-A.

iii. Credits shall be assigned to each course in a programme of study is as follows: L-One hour lecture =One credit ; T- One hour Tutorial=One credit ; P- Two hours of laboratory /Seminar = One credit ; S/A- Four hours of Self- study/ Assignments = One credit.

iv. Each course in a programme of study shall be represented as L-T-P-S-C , where L, T, P, S, and C means respectively, number of lecture hours per week , number of tutorial hours per week , number of laboratory /seminar hours per week , number of self- study hours per week, and the number of credits assigned to the course.

v. **A course shall have either or all the four components**. Consider the Following example; (1) A course may have only lecture component of 4 hours per week, then it will be represented ,as 4:0:0:0:4 . (2) A course may have 3 hours of lecture and one hour of tutorial, then it will be represented as 3:1:0:0:4. (3) If, the course, has only laboratory component of 2 hours duration and one hour of tutorial, then it will be represented, as 0:1:2:0:2. (4) For Self-Study/ Assignments course of 4 hours duration, then it will be repented as 0:0:0:4:1.

vi. The number of credits required to be earned for degree programme shall be calculated at an average of **TWENTY** credits per main semester. For example, a **four** year degree programme shall comprise of **eight** main semesters and therefore require 160 credits, for three year degree programme shall comprise of **six** main semesters and therefore require 120 credits, and for degree programme of **five** years, the number of credits required to be earned shall be 200. For lateral entry, the number of credits required to be earned shall be 120 (for four year degree programme).

A variation of 10% credits is allowed.

vii. A full time student shall normally register for a minimum of **18** credits and maximum of **22** credits during main semester, whereas in supplementary semester a maximum of 12 credits.

viii. Every course in a programme of study normally runs for the full length of a semester.

4. ADMISSION:

Admission to the University shall normally be made at the commencement of each academic year for various programmes of study except research programmes. The date for advertisement, entrance examination, if any, counseling, admission, registration, commencement of classes, and other details for the academic session shall be notified by the Registrar, from time to time.

7. ELIGIBILITY FOR ADMISSION:

The Admission of students to various programmes of studies offered by the University shall fulfill the minimum qualifications laid down by the University, GoK, AICTE, UGC and MHRD for the programme of study concerned, subject to Rules of reservation for candidates belonging to SC, ST, and other Backward Classes as laid down by the State Government from time to time.

8. ADMISSION PROCESS:

Admission process for various programmes shall be as follows:

i. Admission to I year / I semester professional programmes (B.TECH, B.ARCH., BCA, BBA etc.) shall be open to the candidates who have passed the second year PUC or XII standard or Equivalent examination recognized by the University

ii. NRI/PIO/FN seeking admission to the above professional programmes shall apply separately with equivalency/eligibility/migration certificate along with passport/visa/clearance/NOC from concerned bodies to the Admission Committee. Only after the eligibility is ascertained, a NRI/PIO/FN can appear for the entrance test conducted by the University.

iii. A candidate seeking admission under the Government Quota shall follow the procedures of the Common Entrance Test (CET) as notified by the Government of Karnataka from time to time, and NATA/JEE ranks for B.Arch.

iv. A candidate seeking admission under the University Quota (Management Quota) shall appear for entrance test conducted by the University by submitting application form and paying the prescribed entrance test fee. However, the students who have cleared and obtained rank in KEA/CET/JEE paper I&II/NATA etc., need not write the University Entrance Exam.

v. Admission Committee shall prepare a merit list for each of programme of study subject to a minimum performance criterion in the entrance test as prescribed by the admission committee from time to time, and the percentage of marks obtained in the qualifying examinations as prescribed by Government of Karnataka.

vi. Merit list as prepared by Admission Committee shall be submitted to the Chancellor for his/her approval.

vii. The Admission Committee shall notify the list of selected candidates.

viii. The selected candidates (as per the notification) shall complete the admission process by submitting the requisite forms along with supporting documents, paying the prescribed fees and full filling any other requirements mentioned in the notification.

ix. Candidates who have passed a qualifying examination not conducted by Government of Karnataka or this University shall submit the eligibility and migration certificate in original for admission to a programme of study.

x. Candidate shall be required to submit medical certificate and character certificate from the recognized Doctor and Head of the institution last attended respectively.

xi. Admission to IInd year/ III Sem B.E./B.Tech. under lateral entry scheme shall be open to the candidates who have passed the three year diploma from the Karnataka state and secured not less than 45% of marks in aggregate (considering the marks of all six semesters). In case of SC/ST and OBC students from Karnataka state the eligibility shall be 40%. However candidates who have passed diploma from other than the Karnataka state shall provide the equivalence/ eligibility certificate from the director of technical education, Bangalore. Also, the students who have passed B.Sc. Degree from the recognized university or equivalent qualification as recognized by university and secured not less than 45% marks in aggregate (considering the marks of all six semesters). In case of SC/ST and OBC students from Karnataka state the eligibility shall be 40% .However candidates who have passed B.Sc. from other than the Karnataka state shall provide the equivalence/ eligibility certificate from the competent authority.

8 A. Mandatory Induction Programme:

All the new entrants to the university shall attend Mandatory Induction Programme for duration of 3 weeks.

9. ATTENDANCE REQUIREMENT:

i. Each semester is considered as one unit and the student is required to have a minimum attendance of 85% in each course with a provision of condonation of 10% of attendance by the Vice- Chancellor on the specific recommendation of the Dean of the Faculty, indicating reasonable cause such as medical ground participation in University level sports, cultural programs, seminars, workshops, paper presentation, etc.

ii. The calculation of the attendance shall be based on the reopening date notified by the University by its calendar of events from time to time. However, for first semester (III semester for later entry) students the same will be reckoned from the date of admission to the course as per KEA CET/the University allotment.

iii. The shortage of attendance shall be informed to the students/Parents by the Dean/Chairman/Coordinator/Teacher concerned periodically to be cautious and to make up the shortage. In case, a student’s class attendance in a course is less than as stipulated by the University, the student is said to have dropped that course and the student has to re-register for the dropped course when the course is offered again by the Department if it is a hard core course. The student may choose the same or any alternate core/elective in case the dropped course is soft core/elective course.

iv. Provided that mere omission by the University to inform the student about the shortage of attendance shall not entitle him/her to appear for examination.

10. ASSESSMENT AND EVALUATION:

The assessment and evaluation of each student shall comprise of two components viz; Continuous Internal Evaluation (CIE) and Semester End Examination (SEE). Equal weightage shall be given for CIE and SEE.

10.1. Continuous Internal Evaluation:

The CIE shall be conducted by the course teacher throughout the semester. The suggested components of CIE for Theory and Laboratory/ Project course are as depicted below in Table-I and Table-II respectively.

Table-I: Suggested components of CIE for Theory

Sl. No	Components	Marks
1	Internal Test-I*	15
2	Internal Test-II*	15
3	Internal Test-III*	15
4	Daily/Regular/ Session wise Seminar/Assignment/Mock Evaluation	35

*Average of Best Two performances of the Internal Tests shall be considered for 15 Marks.

Table-II: Suggested components of CIE for Laboratory/Project

Sl. No	Components	Marks
1	Conduction of experiments / Design and fabrication of the system/ Project.	25
2	Evaluation of Lab/project report	15
3	Mock Evaluation/ Presentation	10

The suggested components of CIE for Seminar, Internship and Final Project course are as depicted below in Table-III, Table-IV and Table-V respectively.

Table-III: Suggested components of CIE for Seminar

Sl. No	Components	Marks
1	Identification of Seminar topic from referred Journals in relevant domain suggested by the guide	20
2	Report on Seminar and Evaluation	40
3	Presentation	40

Table-IV: Suggested components of CIE for Internship

Sl. No	Components	Marks
1	Midterm Presentation on Internship	25
2	Report on Internship	25

Table-V: Suggested components of CIE for Final Project

Sl. No	Components	Marks
1	Project Phase-I Literature Survey / Visit to industries / R & D to finalize the project topic	50
2	Project Phase-II a) Design, Testing and Results analysis b) Presentation c) Thesis Writing	30 10 10

10.1.1 Provision to Drop the Course:

In case a student secures less percentage of marks as prescribed in the course, the student is said to have **dropped** that course, and such a student is not allowed to write SEE in that course**.**

A student has to re-register for the **dropped** course when the course is offered again by the department if it is a hard core course. The student may choose the same or an alternate core/elective, in case the dropped course is soft core/elective course.

A student who is said to have dropped the Internship/project work has to re-register for the same subsequently within the stipulated period.

The details of any dropped course shall not appear in the Grade card.

10.1.2 Provision to withdraw course:

A student can withdraw any course within 10 days from the date of commencement of semester. Whenever a student withdraw a course, he/she has to register for the same course in case it is hard core course, the same course or an alternate course if it is soft core/open elective.

10.1.3 Provision for Appeal:

If a student is not satisfied with the evaluation of CIE, he/she can approach the Grievance Reddressal Cell with the written submission together with all the facts, the assignments, test papers etc which were evaluated. This shall be done before the commencement of SEE. The Grievance Reddressal Cell shall look into the details and if necessary take corrective measures.

10.2. Semester End Examination (SEE):

i. A student, who has complied with the minimum specified attendance in a programme and secured greater than or equal to 50% in CIE, shall register for SEE by paying the prescribed fees. The registration process may be online/offline as notified from time to time by the Registrar Evaluation. The registration of a student shall be liable to be cancelled by the office of the Registrar Evaluation, where disciplinary issues are raised by the concerned Dean of Faculty.

ii. After the last date of registration for SEE, the list of students along with their registered courses shall be released by the office of Registrar Evaluation. A student shall verify the accuracy of his/her particulars in the list and discrepancies, if any, shall be reported to office of Registrar Evaluation within Three days from the date of release.

iii. The office of the Registrar Evaluation shall issue the Admit cards to eligible students based on the SEE list. The Admit card of a student shall be valid only for the SEE for which it is issued. The Admit card of a student shall include (i) recent photograph of the student and (ii) registered courses for SEE with subject codes.

iv. With the specific approval of the Vice-Chancellor/the Chancellor, under extra ordinary circumstances, a student whose name does not find place in the student list may be permitted to appear for SEE. The result of such a student may be announced after due verification.

v. The Registrar Evaluation shall appoint Chief Superintendent and Deputy Chief Superintendent for the conduct of SEE as per the Time Table notified.

Theory Examination: The SEE shall be of three hours duration or as mentioned in the scheme. The evaluation for this component shall be 50% of the maximum marks.

Laboratory Examination: The SEE shall be of three hours duration or as mentioned in the scheme and shall comprise of Conduction of experiments / Design and fabrication of the system/ Project. The evaluation for this component shall be 50% of the maximum marks.

The SEE for Laboratory shall be held in batches over several days. There shall be one Internal and one External Examiner and the evaluation shall be based on experimental procedure, write up, coding, execuation, Demonstration, Result analysis / Graphs if any and Viva-voce.

Project Examination: The SEE for the Project shall be evaluated by two examiners jointly and the evaluation shall be based on various components such as, Writing of Abstract, Project report, Oral Presentation, Demonstration and Viva-voce.

Note: The distribution of marks for various components shall be made available to the Examiners by the Registrar Evaluation from time to time.

Question paper pattern: The question paper for theory courses consist of Five modules. In each module, there are two full questions. The Students are required to answer five full questions selecting one from each module.

Note: Some courses which include design, drawing and mandatory courses shall have their own pattern.

Valuation of Answer Scripts:

The Registrar Evaluation shall appoint Chief Coordinator and Deputy Chief Coordinator for the evaluation of SEE answer scripts.

The Registrar Evaluation shall notify the guidelines for the evaluation of various subjects.

The answer books of SEE may be coded before issuing for evaluation by the office of the Registrar Evaluation.

11. ELIGIBILITY FOR PASSING:

The CIE and SEE have equal weightage and the student performance is judged by taking into accounts the results of CIE and SEE individually and also combined. The passing standards are as depicted in the Table-VI.

Table-VI. Eligibility for passing.

	Eligibility for passing.
CIE	≥ 50% of Maximum marks
SEE	≥ 40% of Maximum marks
CIE + SEE	≥ 45% of Maximum marks taken together

The student who passes a course of a semester shall not be allowed to appear for the same again, unless he/she opts for rejection of results as per the following:

i. A student may, at his/her desire, reject his/her total performance of SEE (including CIE marks) or he/she may reject the performance of SEE only. The rejection is permitted only once during the entire course of study.

ii. The student who desires to reject the performance as per (i) shall reject performance in all the courses of that semester, irrespective of whether the student has passed or failed in any course. However, the rejection of performance of 4th year project work shall not be permitted.

iii. A student who desires to reject the total performance of the semester (including CIE), has to take readmission for the relevant semester. Application for such readmission shall be sent to the Registrar through the Dean of faculty within 30 days from the date of announcement of the results. Late submission of application shall not be accepted for any reasons. Readmission to First semester in such cases shall not be considered as Fresh admission.

iv. The student, who desires to reject only the results of SEE of a semester and does not desire readmission, shall be permitted to re-appear for examinations of all the courses of the semester in the subsequent examinations. However, the CIE marks obtained by the student in the rejected semester shall be retained. Application for such readmission shall be sent to the Registrar through the Dean of faculty within 30 days from the date of announcement of the results as per the admission notification issued by the University from time to time. Late submission of application shall not be accepted for any reasons.

Grace Marks: Grace marks shall be awarded to the students in SEE for passing theory/ Laboratory and / or passing semester as per the following attributes:

i. Grace marks shall be awarded to theory / laboratory to a maximum of 2% of total SEE marks, if and only if the student clears that theory / laboratory with minimum prescribed marks.

ii. If a student failed in any one theory / laboratory, he/she is eligible for 3 grace marks, if and only if he/she passes the semester.

A student is granted either i or ii of the above, not both. The granted marks shall be documented in the records but not disclosed in the grade card.

Make Up Examination:

The Make Up examination shall be available to students who may have missed to attend the SEE of one or more courses in a semester for valid reasons and given the ‘I’ grade. The students having ‘X’ grade shall also be eligible to take up Make Up examination. The Make Up examinations shall be held as per dates notified in the Academic Calendar by notification from time to time. The standard of the Make Up examination shall be same as that of regular SEE for the courses.

12. ELIGIBILITY REQUIREMENTS FOR PROMOTION TO NEXT ACADEMIC YEAR:

i. There shall not be any restrictions for promoting from an ODD semester to the next EVEN semester, provided that, the student has fulfilled the attendance requirement.

ii. For vertical promotion in order to move from one academic year to next academic year i.e., from EVEN to ODD semester, a student can carry a maximum of five heads as ‘F’ grades not exceeding a maximum of 14 credits and he/she should maintain a CGPA of 5.

iii. A student who has not obtained the eligibility even after **two/three/four** academic years for a programme of **three/ four / five** years respectively, from the date of admission to first semester **shall discontinue the programme or get readmitted** to first semester as a fresh admission.

iv. The mandatory non credit courses Additional mathematics I & II prescribed at 3rd & 4th semester respectively to lateral entry diploma holders admitted to 3rd semester of B.Tech programs, shall attend the classes during respective semester to complete CIE and attendance requirements and to appear SEE examination. In case any student fail to satisfy the course requirements he / she shall be deemed to have secure F grade. In such case, the student have to fulfill the requirements during subsequent semester/s to appear for SEE.

v. Completion of Additional Mathematics I and II, shall be mandatory for the award of B.Tech. degree.

vi. Lateral entry students with **B.Sc degree** shall clear non credit courses such as Engineering Graphics, Elements of Civil Engineering etc. or as decided by BOS from time to time of the first year engineering programme for the award of degree.

vii. Completion of mandatory non credit courses (as mentioned in vi) shall be mandatory for the award of B.Tech. degree

13. MAXIMUM DURATION FOR UG PROGRAM COMPLETION:

The student shall complete the UG program of **Three/Four/Five** years within a maximum period of **Six/Eight/Ten** Academic years from the date of first admission, failing which he/she shall be declared as **Not Fit for Professional Education.**

14. TYPES OF COURSES:

The curriculum shall be designed based on the concept of **Outcome Based Education**.

The **CBCS** provides choice for the students, to select from the prescribed courses of the programme of study.

i. Different Courses to be offered in a programme of study shall be categorized into the following **SIX** types:

ii. **Humanities and Social Sciences (HSS):** These courses enable the students to acquire the required skills and knowledge essential to pursue a given programme of study. These courses include communication, economics, environment, professional ethics, constitution of India etc;. These courses shall be in the range of 3-6% of the total minimum credits for a programme of study.

iii. **Foundation Courses (Exclusively for Faculty of Engineering & Technology):** Foundation Courses are categorized in to Two parts, (1) Basic Sciences (BS) and (2) Engineering Sciences (ES).

BS courses includes, physics, chemistry, maths, statistics and they are mandatory for all the engineering programme of study.

ES Courses includes, elements of: civil, mechanical, electrical, electronic, engineering and computer programming skills, etc; and they are mandatory for all the engineering programme of study.

These courses shall be in the range of 25-30% of the total minimum credits for a programme of study.

iv. **Core Courses:** Core Courses constitute the core of the programme of the study. The core courses of study are of Two types, VIZ; **(1) Hard Core Course (HCC) and (2) Soft Core Course (SCC).**

Hard Core Course(HCC): The Hard Core Course is a core course in the main programme of study and the students have to study compulsorily. These courses shall be in the range of 25-30% of the total minimum credits for a programme of study.

Soft Core Course(SCC): A core course may be soft core if there is a choice for the student to choose a course from the programme of study or from a sister/ related programme of study which supports the main programme of study. These courses shall be in the range of 2-3% of the total minimum credits for a programme of study.

v. **Elective Courses (EC):** Elective course is a course, which can be chosen from a pool of courses, and which may be very specific or specialized or advanced or supportive to the programme of study or which provides an extended scope or which enables an exposure to some other programme of study or nurtures the students proficiency . Elective courses may be offered by the main programme of study/ related programme of study/sister programme of study, which supports the main programme of study. These courses shall be in the range of 10-20% of the total minimum credits for a programme of study.

Open Elective Course (OEC): An elective course chosen generally from the other programme of study, with an intention to seek exposure is called an **open elective course.** These courses shall be in the range of 5-7% of the total minimum credits for a programme of study.

Self-Study Elective Course (SEC): An elective course designed to acquire an advanced knowledge to support a mini project work or major project work, and a student studies such a course on his own with an advisory support by a teacher is called a **self-study elective course.** These courses shall be in the range of 1-2% of the total minimum credits for a programme of study.

vi. **Audit Courses(AC):** A student may be permitted to take any number of audit courses over and above the graduation requirements for learning a subject.

vii. **Internship, Research or Seminar and Project Work (PW):** These are intended to enhance the student's practical knowledge and exposure to research and industry. The credits for this category shall not exceed 10-12% of the total minimum credits for a programme. Major project work shall normally be carried out in regular semesters.

Internship: The student of UG Programme shall undergo Internship of 8 weeks, preferably, before the commencement of final academic year, whereas for PG Programme they shall undergo Internship of 16 weeks, preferably, at the beginning of third semester.

Project work: For UG programme, a batch of students not more than **four** , shall undertake the innovative project ,preferably, in the final semester and execute in the same semester. For PG programme, project work shall be executed individually by the student in the final semester.

Seminar: Each student shall chose seminar topic on the emerging area only.

viii. **C**ertain programmes of study may have additional requirements such as apprenticeship and residency.

ix. An additional non-credit **summer project** of three weeks duration after the end of every academic year (preferably in the month of August) shall be carried out by all the students.

x. Completion of all summer projects shall be mandatory for the award of B.Tech. degree.

15. GRADING PATTERN:

i. The **SHARNBASVA UNIVERSITY** adopts absolute grading system wherein the marks are converted to grades and every *semeste*r results shall be given with ***Semester Grade Point Average*** **(SGPA)** and ***Cumulative Grade Point Average (CGPA).***

ii. The Grading pattern shall have the letter grade points, as per the following table:

Table –VII. Grades and Grade Points

Level	Outstanding	Excellent	Very Good	Good	Above Average	Average	Poor	Fail
Letter Grade	O	S	A	B	C	D	E	F
Grade Points	10	9	8	7	6	5	4	00

iii. A student shall be awarded Grade F if he/she either fails in the course or is absent for the SEE of that course and the student shall be required to reappear for the semester end examination. If the course is laboratory/practical component, the student shall re-appear both CIE and SEE. Absenting in any one or both of them shall result in award of F Grade.

Table-VIII. Grade Point Scale

Level	Outstanding	Excellent	Very Good	Good	Above Average	Average	Poor	Fail
Letter Grade	O	S	A	B	C	D	E	F
Grade Points	10	9	8	7	6	5	4	00
Score (marks) Range (%)	>=90	<90 >=80	<80 >=70	<70 >=60	<60 >=55	<55 >=50	<50 >=45	<45

iv. **W , X and I Grades:**

W Grade shall be awarded to a student who has withdrawn from a course. Further, this grade shall be recorded in the grade card. If the course is audit course, then there shall be no mention of course in the grade card.

X Grade shall be awarded to a student whose attendance is satisfactory and CIE rating (≥ 60%) in a course, but SEE performance observed to be Poor, for such course X grade shall be awarded. The student shall be provided with an opportunity in the Make-Up examination; however the grades (‘B’ to ‘O’) will be reduced to the next lower grade and the other grades remains same.

I Grade shall be awarded temporarily to a student who is unable to appear for **SEE** for one or more courses, with the permission of the Vice-Chancellor in response to a written appeal by the student, due to valid reasons such as medical emergency, calamity in the family or any other valid reason. For such a student, the I grade shall be converted in to one of the other letter grades as in the table after the completion of scheduled make up SEE. If the student does not appear to the make-up SEE, the I grade shall be converted to an F grade.

v. **AP and AF Grades:** A student shall be awarded either an **Audit Pass (AP) or Audit Fail (AF)** grade for an audit course. The Audit Pass *(AP*) grade shall be awarded if the student satisfies the attendance and performance criteria specified for the course by the concerned Faculty. Otherwise, an *AF* grade shall be awarded.

vi. **COMPUTATION OF SGPA and CGPA:**

COMPUTATION OF SGPA:

Illustration of Computation of SGPA and Format for Transcripts

Computation of SGPA

Illustration No. 1

Course	Credit (C)	Letter Grade	Grade point (G)	Credit point (C X G)
Course 1	4	A	8	4X8=32
Course 2	4	C	6	4X6=24
Course 3	4	B	7	4X7=28
Course 4	3	O	10	3X10=30
Course 5	3	D	5	3X5=15
Course 6	1	C	6	1X6=06
Course 7	1	S	9	1X9=09
Course 8	1	C	6	1X6=06
	21			150

Thus. SGPA=150/21 =7.14

Illustration No. 2

Course	Credit (C)	Letter Grade	Grade point (G)	Credit point (C X G)
Course 1	4	A	8	4X8=32
Course 2	4	C	6	4X6=24
Course 3	4	B	7	4X7=28
Course 4	3	O	10	3X10=30

Course 5	**3**	**F**	**0**	**3X0=0**
Course 6	**1**	**C**	**6**	**1X6=06**
Course 7	**1**	**S**	**9**	**1X9=09**
Course 8	**1**	**C**	**6**	**1X6=06**
	21			**135**

Thus. SGPA=135/21= 6.43

Illustration No. 2(A)

Course	**Credit (C)**	**Letter Grade**	**Grade point (G)**	**Credit point (C X G)**
Course 5	**3**	**B**	**7**	**3X7=21**
	21			**(First Attempt=135+Subsequent Attempt =21)=156**

Thus. SGPA=156/21= 7.43

Illustration No. 3

Course	**Credit (C)**	**Letter Grade**	**Grade point (G)**	**Credit point (C X G)**
Course 1	**4**	**A**	**8**	**4X8=32**
Course 2	**4**	**C**	**6**	**4X6=24**
Course 3	**4**	**B**	**7**	**4X7=28**
Course 4	**3**	**O**	**10**	**3X10=30**
Course 5	**3**	**S**	**9**	**3X9=27**
Course 6	**1**	**C**	**6**	**1X6=06**

Course 7	**1**	**S**	**9**	**1X9=09**
Course 8	**1**	**C**	**6**	**1X6=06**
	21			**162**

Thus. SGPA=162/21= 7.71

CGPA=(21X7.14+21X7.43)/42 =7.28

CGPA after Final Semester

Semester 1	**Semester 2**	**Semester 3**	**Semester 4**	**Semester 5**	**Semester 6**	**Semester 7**	**Semester 8**
Credit : 21 SGPA: 7	**Credit : 21 SGPA: 8.5**	**Credit : 21 SGPA: 9.2**	**Credit : 21 SGPA: 6.86**	**Credit : 21 SGPA: 8.18**	**Credit : 21 SGPA: 7.73**	**Credit : 17 SGPA: 8.68**	**Credit : 17 SGPA: 9.4**

Thus, CGPA= (21X7+21X8.5+21X9.2+21X6.86+21X8.18+21X7.73+17X8.68+17X9.4)/160 = 8.15

16. CONVERSION OF GRADES INTO PERCENTAGE:

Conversion of GPA into Percentage is given below

Percentage of Marks= (CGPA Earend-0.75)*10

CLASS/DIVISION DECLARATION:

Equivalent Percentage	**Class**
>= 70%	**First Class with Distinction**
60% <= % < 70%	**First Class**
50% <= % < 60%	**Second Class**
45% <= % < 50%	**Pass Class**
< 45%	**Fail**

17. AWARD OF PRIZES, MEDALS AND RANKS:

i. For the award of *Prizes* and *Medals*, the conditions stipulated by the Donor shall be considered subject to the provisions of the statutes framed by the University for such awards.

ii. For award of rank in a specialization of B.Tech., the CGPA secured by the student from III to VIII semester is considered, for B.Arch. the CGPA of III to X semester shall be considered, for BCA and BBA all semesters CGPA shall be considered.

iii. A student shall eligible for a rank at the time of award of the degree of B.Tech/BCA/BBA etc, provided the student:

  a. Has passed all semester in all courses in first attempt only in case of candidate admitted to I year.

  b. Has passed III to last semester in all courses in first attempt only in case of candidate admitted under lateral entry scheme.

  c. Has completed all the prescribed Audit/mandatory courses.

  d. Is not repeated in any semester because of rejection of result of a semester/ shortage of attendance etc;.

  e. Has completed all the semester without any break/discontinuity.

  f. Has completed all the semester (I to last semester or III to last semester for lateral entry students) in the University

  g. Has not been transferred from any autonomous institution or from any other University to the Sharnbasva University.

iv. Total number of ranks awarded shall be 10% of the total number of students appeared in final semester subject to the maximum of 10 ranks in a specialization.

v. For award the rank in specialization, a minimum of 10 students should have appeared in the final semester examination.

Illustration.

  a. If 1228 students appeared for the VIII semester in Electronics and Communication Engineering Programme, the number of ranks to be awarded for Electronics and Communication Engineering shall be 10.

  b. If 90 students appeared for the VIII semester in Biomedical Engineering, the number of ranks to be awarded for Biomedical Engineering will be 9.

  c. If 10 or less students appeared for the final semester of any degree, the number of ranks shall be awarded is one.

vi. In case of fraction number of ranks, it is rounded to higher integer when the first decimal place value is greater than or equal to 5.

vii. Ranks are awarded based on the merit of the students as determined CGPA. If two or more students get the same CGPA, the tie shall be resolved by considering the number of times student has obtained higher SGPA. If it is not resolved even at this stage, the number of times student has obtained higher grades like S, A, B, etc., shall be taken into account to decide the order of the rank.

18. APPLICABILITY AND POWER TO MODIFY:

i. The regulations governing the degree of Bachelor of Engineering / Technology of SUK shall be binding on all concerened.

ii. Not withstanding any thing contained in the foregoing the university shall have the power to issue directions /orders to address any difficulty.

iii. Nothing in the foregoing may be construed as limiting the power of the University to amend, modify or repeal any or all of the above.

ENGINEERING MATHEMATICS-III

(Common to all branches)

[As per Choice Based Credit System (CBCS) scheme]

(Effective from the academic year 2019-20)

SEMESTER-III

Course Code :	**19MAT31**	**CIE Marks :**	**50**
Contact Hours/Week :	**04**	**SEE Marks:**	**50**
Total Hours:	**50**	**Exam Hours:**	**03**

Credits: 04

Course Learning Objectives:

This course will enable students to:

- Introduce most commonly used analytical and numerical methods in the different engineering fields.
- Learn Laplace transform and Z-transforms, statistical methods, numerical methods.
- Solve the problem on Interpolation.
- To discuss the random variable and associated probability distributions.

Modules	**Teaching Hours**	**Revised Bloom's Taxonomy (RBT) Levels**
Module - 1		
LAPLACE TRANSFORMS : Definition, Transforms of Elementary functions, properties, periodic function, Unit step function, Unit impulse function. **INVERSE LAPLACE TRANSFORMS :** Definition, Convolution Theorem(without proof), Finding Inverse Laplace transform by convolution Theorem. Solution of Linear Differential equations using Laplace Transforms and Applications(5 Assignment Problem).	**10 - Hours**	**L1 & L2**
Module - 2		
Z- **TRANSFORMS:** Difference Equations ,Basic definitions, Damping rule, Shifting rule, Initial and Final Value theorems(without proof) and problems. Inverse Z-transforms. Applications of Z-transforms to solve difference equation(5 Assignment Problem).	**10 - Hours**	**L1 & L2**
Module - 3		
STATISTICAL METHODS: Correlation-karl Pearson's co-efficient of correlation problems.Regression analysis lines of regression (without proof)-problems. **CURVE FITTING:** Curve fitting by the method of least square. Fitting of the curves of the form $y = ax + b, y = ax^2 + bx + c$ & $y = ae^{bx}$. **Numerical Methods:** Numerical solution of algebraic and trans cendental equations by Regula - Falsi Method and Newton-Raphson method. (5 Assignment Problem).	**10 - Hours**	**L1 & L2**
Module - 4		
FINITE DIFFERENCE: Forward and Backward differences,	**10 - Hours**	**L1 & L2**

Newton's forward and backward interpolation formulae.Divided difference-Newton's divided difference formulae.Lagrange's-interpolation formula and inverse interpolation formula(all formula without proof) problems. **NUMERICAL INTEGRATION:**Simpsons$(\frac{1}{3})^{rd}$, $(\frac{3}{8})^{th}$ rules, Weddle's rule (without proof) problems.(5 Assignment Problem).		
Module - 5		
Probability Distribution: Random variables(discrete and continuous) probability mass/density functions. Binomial distribution, Poisson distribution. Exponential and Normal distributions. Problems.(5 Assignment Problem).	**10 - Hours**	**L3**
Course outcomes: On completion of this course, students are able to: • Know the use of Laplace transform and inverse Laplace transform in signal and image processing. • Explain the general linear system theory for continuous time signals and digital signal processing using the Z-transform. • Employ appropriate numerical methods to solve algebraic and transcendental equations. • Apply Green's Theorem, Divergence Theorem and Stokes' theorem in various application in the field of electro-magnetic and gravitational fields and fluid flow problems.		
Question paper pattern: • The question paper will have ten questions. • Each full Question consisting of 16 marks • There will be 2 full questions (with a maximum of four sub questions) from each module. • Each full question will have sub questions covering all the topics under a module. • The students will have to answer 5 full questions, selecting one full question from each module.		
Text Books: *1. B.S. Grewal: Higher Engineering Mathematics, Khanna Publishers, 43rd Ed., 2015.* *2. E. Kreyszig: Advanced Engineering Mathematics, John Wiley & Sons,10th Ed., 2015.*		
Reference Books: *1. N.P.Bali and Manish Goyal: A Text Book of Engineering Mathematics, Laxmi Publishers , 7th Ed., 2010.* *2. B.V.Ramana: "Higher Engineering Mathematics" Tata McGraw-Hill, 2006.* *3. H. K. Dass and Er. Rajnish Verma: "Higher Engineering Mathematics", S. Chand publishing, 1st edition, 2011.*		
Web Link and Video Lectures: *1. http://nptel.ac.in/courses.php?disciplineID=111* *2. http://www.khanacademy.org/* *3. http://www.class-central.com/subject/math*		

DATA STRUCTURES IN C AND APPLICATIONS

[As per Choice Based Credit System (CBCS) scheme]

(Effective from the academic year 2018-2019)

SEMESTER – III

Subject Code	18CS/IS32	**CIE Marks**	50
Number of Lecture Hours/Week	04	**SEE Marks**	50
Total Number of Lecture Hours	48	**Exam Hours**	03

CREDITS – 04

Course objectives: This course will enable students

- To impart the basic concepts of data structures and algorithms.
- To understand concepts about searching and sorting techniques
- To understand basic concepts about stacks, queues, lists, trees and graphs.
- To enable them to write algorithms for solving problems with the help of fundamental data structures

Module I	**Teaching Hours**
Introduction: Data Structures, Data structure Operations, Pointers and Dynamic Memory Allocation, Data Abstraction. **Arrays and structures** :Arrays, dynamic allocated arrays ,structures & unions, Polynomials and Sparse Matrices. **Array Operations:** Traversing, inserting, deleting, searching, and sorting. **Strings**: Basic Terminology, Storing, Programming Examples.	10
Module II	
Stacks & Queues : **Stacks**: Definition, Stack Operations, Array Representation of Stacks, Stack Applications: Polish notation, Infix to postfix conversion, evaluation of postfix expression, **Recursion**: Factorial, Fibonacci Sequence, Tower of Hanoi, Ackerman's function. **Queues:** Definition, Representation-array & linked representation of queues. Queue Operations, Circular Queues, Dequeues, Priority Queues.	10
Module III	
Linked Lists: Definition, Representation of linked lists in Memory, Memory allocation; Linked list operations: Traversing, Searching, Insertion, and Deletion. Doubly Linked lists and header linked lists. Linked Stacks and Queues. Applications of Linked lists – Polynomials, Additional list operations-inverting singly linked list, concatenating singly linked list. Sparse matrix representation.	08
Module IV	
Trees: Definition, Representation of trees, Binary Trees, Properties of Binary trees, Array and linked Representation of Binary Trees, Binary Tree Traversals - Inorder, postorder, preorder; Additional Binary tree operations-copying binary tree, testing equality. Threaded binary trees, Binary Search Trees – Definition, Insertion, Deletion, Traversal, Searching.	10
Module V	
Graphs: Definitions, Terminologies, Matrix and Adjacency List Representation Of Graphs, Elementary Graph operations, Traversal methods: Breadth First Search and Depth First Search. **Sorting and Searching:** Insertion Sort, Radix sort, selection sort. **Hashing**: Hash Table organizations, Hashing Functions.	10

Course Outcomes

After studying this course, students will be able to:

CO 1: For a given Search problem (Linear Search and Binary Search) student will able to implement it.

CO 2: For a given problem of Stacks, Queues and linked lists student will be able to implement its applications.

CO 3: Students will be able to write functions on different types of trees and their operations.

CO 4: Student will be able to implement Graphs, Searching, Sorting, Hashing and their applications.

.

Question paper pattern:

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

Text Books:

1. Fundamentals of Data Structures in C - Ellis Horowitz and Sartaj Sahni, 2nd edition, Universities Press, 2014.
2. Data Structures - Seymour Lipschutz, Schaum's Outlines, Revised 1st edition, McGraw Hill, 2014

Reference Books:

1. Data Structures: A Pseudo-code approach with C –Gilberg & Forouzan, 2nd edition, Cengage Learning, 2014.
2. Data Structures using C, , Reema Thareja, 3rd edition Oxford press, 2012.
3. An Introduction to Data Structures with Applications- Jean-Paul Tremblay & Paul G. Sorenson, 2nd Edition, McGraw Hill, 2013.
4. Data Structures using C - A M Tenenbaum, PHI, 1989.
5. Data Structures and Program Design in C - Robert Kruse, 2nd edition, PHI, 1996.

ELECTRONICS CIRCUITS AND LOGIC DESIGN

[As per Choice Based Credit System (CBCS) scheme]

(Effective from the academic year 2018-2019)

SEMESTER – III

Subject Code	18CS/IS33	**CIE Marks**	50
Number of Lecture Hours/Week	04	**SEE Marks**	50
Total Number of Lecture Hours	48	**Exam Hours**	03

CREDITS - 04

Course objectives: This course will enable students to

- Recall and Recognize construction and characteristics of JFETs and MOSFETs and differentiate with BJT
- Demonstrate and Analyze Operational Amplifier circuits and their applications
- Describe, Illustrate and Analyze Combinational Logic circuits, Simplification of Algebraic Equations using Karnaugh Maps and Quine McClusky Techniques.
- Describe and Design Decoders, Encoders, Digital multiplexers, Adders and Subtractors, Binary comparators, Latches and Master-Slave Flip-Flops.
- Describe, Design and Analyze Synchronous and Asynchronous Sequential.
- Explain and design registers and Counters, A/D and D/A converters.

Module I	**Teaching Hours**
Field Effect Transistors: Junction Field Effect Transistors, MOSFETs, Differences between JFETs and MOSFETs, Biasing MOSFETs, FET Applications, CMOS Devices. Wave-Shaping Circuits: Integrated Circuit(IC) Multivibrators. Introduction to Operational Amplifier: Ideal v/s practical Opamp, Performance Parameters,.	10
Module II	
The Basic Gates: Review of Basic Logic gates, positive and negative logic Introduction to HDL. Combinational Logic Circuits: Sum-of-Products Method, Truth Table to Karnaugh Map, Pairs, Quads, and Octets, Karnaugh Simplifications, Don’t-care Conditions, Product-of-sums Method, Product-of-sums simplifications, Simplification by QuineMcClusky Method, Hazards and Hazard covers, HDL Implementation Models.	10
Module III	
Data-Processing Circuits: Multiplexers, Demultiplexers, 1-of-16 Decoder, BCD to Decimal Decoders, Seven Segment Decoders, Encoders, Exclusive-OR Gates, Parity Generators and Checkers, Magnitude Comparator, Programmable Array Logic, Programmable Logic Arrays, HDL Implementation of Data Processing Circuits. Arithmetic Building Blocks, Arithmetic Logic Unit Flip- Flops: RS Flip-Flops, Gated Flip-Flops, Edge-triggered RS FLIP-FLOP, Edge-triggered D FLIP-FLOPs, Edge-triggered JK FLIPFLOPs.	08
Module IV	
Flip- Flops: FLIP-FLOP Timing, JK Master-slave FLIP-FLOP, Switch Contact Bounce Circuits, Various Representation of FLIP-FLOPs, HDL Implementation of FLIP-FLOP. Registers: Types of Registers, Serial In - Serial Out, Serial In - Parallel out, Parallel In - Serial Out, Parallel In - Parallel Out, Universal Shift Register, Applications of Shift Registers, Register implementation in HDL. Counters: Asynchronous Counters, Decoding Gates,3s.	10
Module V	
Design of synchronous and asynchronous sequential circuits: model selection, state transition diagram, state synthesis table design equation and circuit diagram, implementation using read only memory. **D/A Conversion and A/D Conversion:** Variable, Resistor Networks, Binary Ladders, D/A Converters, D/A Accuracy and Resolution, A/D Converter-Simultaneous Conversion, A/D Converter-Counter Method, Continuous A/D Conversion, A/D Techniques, Dual-slope A/D Conversion .	10

Course Outcomes

After Studying this course, students will be able to

CO 1: Acquire knowledge of

- JFETs and MOSFETs , Operational Amplifier circuits and their applications
- Combinational Logic, Simplification Techniques using Karnaugh Maps, Quine McClusky technique.
- Operation of Decoders, Encoders, Multiplexers, Adders and Subtractors.

CO 2: Working of Latches, Flip-Flops, Designing Registers, Counters, A/D and D/A Converters.

CO 3: Analyze the performance of

- JFETs and MOSFETs , Operational Amplifier circuits
- Simplification Techniques using Karnaugh Maps, Quine McClusky Technique.
- Synchronous and Asynchronous Sequential Circuits.

CO 4: Apply the knowledge gained in the design of Counters, Registers and A/D & D/A converters

Question paper pattern:

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

Text Books:

1. Anil K Maini, Varsha Agarwal: Electronic Devices and Circuits, Wiley, 2012.
2. Donald P Leach, Albert Paul Malvino & Goutam Saha: Digital Principles and Applications, 8th Edition, Tata McGraw Hill, 2015

Reference Books:

1. Stephen Brown, Zvonko Vranesic: Fundamentals of Digital Logic Design with VHDL, 2nd Edition,Tata McGraw Hill, 2005.
2. R D Sudhaker Samuel: Illustrative Approach to Logic Design, Sanguine-Pearson, 2010.
3. M Morris Mano: Digital Logic and Computer Design, 10 th Edition, Pearson, 2008

Computer Organization & Architecture

[As per Choice Based Credit System (CBCS) scheme]
(Effective from the academic year 2018-2019)
SEMESTER – III

Subject Code	18CS/IS34	**CIE Marks**	50
Number of Lecture Hours/Week	04	**SEE Marks**	50
Total Number of Lecture Hours	48	**Exam Hours**	03

CREDITS – 04

Course objectives: This course will enable students

- How Computer Systems work & the basic principles
- Instruction Level Architecture and Instruction Execution.
- The current state of art in memory system design
- How I/O devices are accessed and its principles.
  To provide the knowledge on Instruction Level Parallelism.
- To impart the knowledge on micro programming.
- Understand Concepts of advanced pipelining techniques, Computer Arithmetic and parallel processing

Module I	**Teaching Hours**
Functional blocks of a computer: Functional units, Basic operational concepts, Bus Structure, Software, and Performance. Signed number representation, character representation. Memory location and address, Instruction and sequencing, Basic IO operations, Addressing Modes, Additional Instructions: Shift and Rotate Instructions	10
Module II	
Basic Processing Unit: Single Bus Organization, Multiple Bus Organization, Hardwired and micro-programmed design approaches. **Input Output Organization**: Accessing I/O devices, Interrupts, DMA, Buses.	10
Module III	
The Memory System: Semiconductor RAM memories (SDRAM, ADRAM), Cache Memories, Performance Consideration. **Arithmetic:** Addition and Subtraction of Signed Numbers, Design of Fast Adders, Multiplication of Positive Numbers, Signed Operand Multiplication, Fast Multiplication.	10
Module IV	
Pipelining: Introduction, Major Hurdles of Pipelining, How is pipelining implemented?, What makes pipeline hard to implement, Instruction Level Parallelism: Concepts and Challenges.	10
Module V	
Memory Hierarchy: Introduction, Cache Performance, Six basic Cache Optimization, Virtual Memory, Memory Hierarchy Design: 10 Advanced optimizations of cache performances.	08

Course Outcomes

After studying this course, students will be able to:
CO1: Identify basic structure of computer and its performance measures.
CO2: Demonstrate functioning of different sub systems, such as processor, Input/output, and memory.
CO3: Understand the concepts of Pipelining and parallel processor.
CO4: Understand the concepts of Memory organization.
CO5: Understand the concepts of, Arithmetic Operations and Characters.

Question paper pattern:

- The question paper will have ten questions.
- There will be 2 questions from each module.

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

Text Books:

1. Carl Hamacher, Z. Vranesic & S.Zaky, “Computer Organization”,5th Edition , Tata McGraw-Hill Publishing Company Ltd. New Delhi, 2002.
2. John L. Hennessy and David A. Patterson, Computer Architecture: A quantitative approach, 5th edition, Morgan Kaufmann Elseveir, 2013

Reference Books:

1. Morris Mano, “Computer System Architecture”, PHI, 19862.William Stallings Computer Organization & Architecture, 7th Edition , PHI 2006.
2. Kai Hwang and Naresh Jotwani, Advanced Computer Architecture (SIE): Parallelism, Scalability, Programmability, McGraw Hill Education 3/e. 2015.

Data Structures Lab

[As per Choice Based Credit System (CBCS) scheme]

(Effective from the academic year 2018-2019)

SEMESTER – III

Subject Code	18CSL/ISL35	**CIE Marks**	50
Number of Lecture Hours/Week	02	**SEE Marks**	50
Total Number of Lecture Hours	48	**Exam Hours**	03

CREDITS - 01

Course objectives: This course will enable students

- To design, develop, test and debug in C/C++ language considering appropriate data structure.
- Illustrate and implement data types such as stack, queue and linked list and apply them for the given problem.
- Illustrate and implement the trees and other data structures.

PART-A

Students are required to implement following programs using C/C++.

1. Implementation of stack ADT using arrays
2. Implementation of queue ADT using arrays
3. Implementation of List ADT
4. Implementation of Graph ADT using List
5. Implementation of tree ADT using List / Array

Part B

Application of Stack

1. Implementation of Infix to Postfix conversion.
2. Implementation of postfix evaluation.

Application of Queue

3. Implementation of Priority queue program using array.
4. Implementation of multiple stacks and queues

Application of List

5. Implementation of sparse matrix multiplication.
6. Implementation of Linked Lists menu driven program (stack and queue)

Application of Graph & Tree

7. Implementation of construction of expression tree using postfix expression.
8. Implementation of various operations on tree like – copying tree, counting the number of nodes in the tree.
9. Implementation of Binary Heap program

Course Outcomes

After studying this course, students will be able to:

CO 1: Understand and explore the fundamental concepts of various data structures.

CO 2: Analyze and represent various data structures

CO 3: Design algorithms on different data structures like Stack, Queue, List, Tree and hashing.

CO 3: Implement programs with suitable data structure based on the requirements of the application

Electronics Circuits & Logic Design Lab

[As per Choice Based Credit System (CBCS) scheme]
(Effective from the academic year 2018-2019)
SEMESTER – III

Subject Code	18CSL/ISL36	**CIE Marks**	50
Number of Lecture Hours/Week	02	**SEE Marks**	50
Total Number of Lecture Hours	48	**Exam Hours**	03

CREDITS - 01

Course objectives: This course will enable students

PART A

1. A. Design and construct a Schmitt trigger using Op-Amp for given UTP and LTP values and demonstrate its working.
   B. Design and implement a Schmitt trigger using Op-Amp using a simulation package for two sets of UTP and LTP values and demonstrate its working.

2. A. Design and construct a rectangular waveform generator (Op-Amp relaxation oscillator) for given frequency and demonstrate its working.
   B.Design and implement a rectangular waveform generator (Op-Amp relaxation oscillator) using a simulation package and demonstrate the change in frequency when all resistor values are doubled.

3. Design and implement an A stable multi vibrator circuit using 555 timer for a given frequency and duty cycle.

PART B

1. Design and implement Half adder, Full Adder, Half Subtractor, Full Subtractor using basic gates.

2. Given a 4-variable logic expression, simplify it using Entered Variable Map and realize the simplified logic expression using 8:1 multiplexer IC.
3. Design and develop the Verilog /VHDL code for an 8:1 multiplexer. Simulate and verify it’s working.
4. Design and implement code converter
   I)Binary to Gray
   II) Gray to Binary Code using basic gates.

5. Design and verify the Truth Table of 3-bit Parity Generator and 4-bit Parity Checker using basic Logic Gates with an even parity bit.
6. a) Realize a J-K Master / Slave Flip-Flop using NAND gates and verify its truth table.
   b) Design and develop the Verilog / VHDL code for D Flip-Flop with positive edge triggering.

Course Outcomes

After studying this course, students will be able to:
CO 1: Use various Electronic Devices like Cathode ray Oscilloscope, Signal generators, Digital Trainer Kit, Multimeters and components like Resistors, Capacitors, Op amp and Integrated Circuit.
CO2: Design and demonstrate various combinational logic circuits.
CO3: Design and demonstrate various types of counters and Registers using Flip-flops
CO4: Use simulation package to design circuits.
CO5: Understand the working and implementation of ALU.

Unix Shell Programming Lab

[As per Choice Based Credit System (CBCS) scheme]
(Effective from the academic year 2018-2019)
SEMESTER – III

Subject Code	18CSL/ISL37	**CIE Marks**	50
Number of Lecture Hours/Week	02	**SEE Marks**	50
Total Number of Lecture Hours	48	**Exam Hours**	03

CREDITS - 01

Course objectives: This course will enable students

- To Study of UNIX basic Commands
- To introduce Basic Unix general purpose Commands.
- To write shell scripts to solve problems

Part A

1. Study of UNIX basic commands:
   cal, date, echo, printf, bc, script, mailx, passwd, who, uname, tty, stty, pwd, cd, mkdir, rmdir, ls, cat, cp, rm, mv, more, file, wc, od, cmp,comm, diff, chmod.
2. Study of vi editor.
3. Write a script to study if...else, if and case statements.
4. Write a script to study for, while and until.
5. Study the Filters for stream handling features of the shell for input and output.
   E.g. pr, head, tail, cut, paste, sort, nl, uniq, tr.

Part B

1.
   a) Write a Shell program to count number of user's login and print first login user information
   b) Write Shell Script to read user name and find whether the user is currently working in the system or not.

2.
   a) Write shell script for-
   (i) Showing the count of users logged in.
   (ii) Printing Column list of files in your home directory.
   (iii) Listing your job with below normal priority.
   (IV) Continue running your job after logging out.
   b) Write a shell script to create a file. Follow the instructions
   (i) Input a page profile to yourself, copy it into other existing file;
   (ii) Start printing file at certain line.
   (iii) Print all the difference between two file, copy the two files.
   (iv) Print lines matching certain word pattern.

3.
   a) Write a shell script that displays a list of all the files in the current directory to which the user has read, write and execute permissions.
   b) Write a shell script that accepts a file name, starting and ending line numbers as arguments and displays all the lines between the given line numbers.

4.
   a) Write a shell script that receives any number of file names as arguments checks if every argument is a file or directory, when it is a file, report no of lines in it.
   b) Write a shell script that accepts a list of file names as its arguments, count and reports the occurrence of each word that is present in the first argument file on other argument files.

5.
   a) Write a grep/egrep script to find the number of words character, words and lines in a file.
   b) Write an egrep script to display list of files in the directory.

6.

a) Write an awk script to count the number of lines in a file that do not Contain vowels.

b) Write an awk script to find the number of characters, words and lines in a file.

7.

a) Write a Perl script to compute the power of a given number.

b) Write a Perl script to check a number is prime or not.

Course Outcomes

After studying this course, students will be able to:
CO1: Work on any Unix platform with confidence
CO2: Write effective scripts for their day to day jobs
CO3: Understand and use most of the Unix features and commands
CO5: Will be able to do Basic System administration.

ENGINEERING MATHEMATICS-IV

(Common to all branches)

[As per Choice Based Credit System (CBCS) scheme]

(Effective from the academic year 2019-20)

SEMESTER-IV

Course Code :	**19MAT41**	**CIE Marks :**	**50**
Contact Hours/Week :	**04**	**SEE Marks:**	**50**
Total Hours:	**50**	**Exam Hours:**	**03**

Credits: 04

Course Learning Objectives:

This course will enable students to:

- Learn Fourier series and Fourier transforms.
- Conversant with numerical methods to solve ordinary differential equations, complex analysis, joint probability distribution and stochastic processes arising in science and engineering.

Modules	**Teaching Hours**	**Revised Bloom's Taxonomy (RBT) Levels**
Module - 1		
Fourier Series: Periodic functions, Dirichlet's condition, Fourier Series of periodicfunctions with period 2π and with arbitrary period 2c. Fourier series of even and odd functions Half range Fourier Series, practical harmonic analysis(5 Assignment Problem).	**10 - Hours**	**L1 & L2**
Module - 2		
Fourier Transforms: Infinite Fourier transforms, Fourier sine and cosine transforms. Inverse Fourier-transform (5 Assignment Problem). **Complex line Integrals:** Cauchy's Integration theorem, Cauchy integral formula, Laurent's Series, types of singularities. Residue, Poles, Cauchy's Residue theorem (without proof) and Problems. **Transformations:** Bilinear transformations and problems.	**10 - Hours**	**L1 & L2**
Module - 3		
Numerical Methods: Numerical solution of ordinary differential equations of first order and first degree, Taylor's series method, modified Euler's-method Runge Kutta method of fourth order. Milne's and Adams- Bashforth predictor and corrector methods (No derivations of formulae). (5 Assignment Problem).	**10 - Hours**	**L1 & L2**
Module - 4		
Numerical Methods: Numerical solution of second order ordinary differential equations, Runge- Kutta Method and Milne's Method, Numerical solution of P.D.E: Numerical solution of heat equation, wave equation , problems. (5 Assignment Problem).	**10 - Hours**	**L1 & L2**
Module - 5		
Joint probability distribution: Joint Probability distribution for two discrete random variables, expectation, covariance, correlation coefficient	**10 - Hours**	**L3**

Stochastic process: Stochastic processes, probability vector, stochastic matrices, fixed points, regular stochastic matrices, Markov chains, higher transition probability- simple problems.(5 Assignment Problem).		
Course Outcomes: On completion of this course, students are able to: • Know the use of periodic signals and Fourier series to analyze circuits and system communications. • Explain the general linear system theory for continuous time signals and digital signal processing using the Fourier Transform. • Solve first and second order ordinary differential equations arising in flow problems using single step and multistep numerical methods. • Understand the analyticity, potential fields, residues and poles of complex potentials in field theory and electromagnetic theory. • Describe bilinear transformation arising in aerofoil theory, fluid flow visualization and image processing. • Solve problems on probability distributions relating to digital signal processing, information theory and optimization concepts of stability of design and structural engineering. • Determine joint probability distributions and stochastic matrix connected with the multivariable correlation problems for feasible random events. • Define transition probability matrix of a Markov chain and solve problems related to discrete parameter random process.		
Question paper pattern: • The question paper will have ten questions. • Each full Question consisting of 16 marks • There will be 2 full questions (with a maximum of four sub questions) from each module. • Each full question will have sub questions covering all the topics under a module. • The students will have to answer 5 full questions, selecting one full question from each module.		
Text Books: *1. B.S. Grewal: Higher Engineering Mathematics, Khanna Publishers, 43rd Ed., 2015.* *2. E. Kreyszig: Advanced Engineering Mathematics, John Wiley & Sons,10th Ed., 2015.*		
Reference Books: *1. N.P.Bali and Manish Goyal: A Text Book of Engineering Mathematics, Laxmi Publishers , 7th Ed., 2010.* *2. B.V.Ramana: "Higher Engineering Mathematics" Tata McGraw-Hill, 2006.* *3. H. K. Dass and Er. Rajnish Verma: "Higher Engineering Mathematics", S. Chand publishing, 1st edition, 2011.*		
Web Link and Video Lectures: *1. http://nptel.ac.in/courses.php?disciplineID=111* *2. http://www.khanacademy.org/* *3. http://www.class-central.com/subject/math*		

DESIGN AND ANALYSIS OF ALGORITHMS

[As per Choice Based Credit System (CBCS) scheme]
(Effective from the academic year 2018-2019)
SEMESTER – IV

Subject Code	18CS/IS42	**CIE Marks**	50
Number of Lecture Hours/Week	04	**SEE Marks**	50
Total Number of Lecture Hours	48	**Exam Hours**	03

CREDITS - 04

Course objectives: This course will enable students to

- Explain various computational problem solving techniques.
- Apply appropriate method to solve a given problem.
- Describe various methods of algorithm analysis.

Module I	**Teaching Hours**
Introduction: Notion of Algorithm, Review of Asymptotic Notations, Mathematical analysis of Non-Recursive and recursive Algorithms with Examples. Important Problem Types: Sorting, Searching, String processing, Graph Problems, Combinatorial Problems. Fundamental Data Structures: Stacks, Queues, Graphs, Trees, Sets and Dictionaries.	10
Module II	
Divide and Conquer: General method, Binary search, Recurrence equation for divide and conquer, Finding the maximum and minimum , Merge sort, Quick sort , Advantages and Disadvantages of divide and conquer. Decrease and Conquer Approach: Topological Sorting	10
Module III	
Greedy Method: General method, Coin Change Problem, Knapsack Problem, Job sequencing with deadlines. Minimum cost spanning trees: Prim’s Algorithm, Kruskal’s Algorithm . Single source shortest paths: Dijkstra's Algorithm . Optimal Tree problem: Huffman Trees and Codes . Transform and Conquer Approach: Heaps and Heap Sort	08
Module IV	
Dynamic Programming: General method with Examples. Transitive Closure: Warshall’s Algorithm, All Pairs Shortest Paths: Floyd's Algorithm, Knapsack problem,Bellman-Ford Algorithm, Travelling Sales Person problem	10
Module V	
Backtracking: General method , N-Queens problem , Sum of subsets problem , Graph coloring , Hamiltonian cycles . Branch and Bound: Assignment Problem, Travelling Sales Person problem , 0/1 Knapsack problem : LC Branch and Bound solution , FIFO Branch and Bound solution	10

Course Outcomes

After studying this course, students will be able to:

CO 1: Describe computational solution to well known problems like searching, sorting etc.
CO 2: Estimate the computational complexity of different algorithms.
CO 3: Devise an algorithm using appropriate design strategies for problem solving.

Question paper pattern:

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

Text Books:

1. Introduction to the Design and Analysis of Algorithms, Anany Levitin:, 2nd Edition, 2009. Pearson.

2. Computer Algorithms/C++, Ellis Horowitz, Satraj Sahni and Rajasekaran, 2nd Edition, 2014, Universities Press/

Reference Books:

1. Introduction to Algorithms, Thomas H. Cormen, Charles E. Leiserson, Ronal L. Rivest, Clifford
   Stein, 3rd Edition, PHI.
2. Design and Analysis of Algorithms, S. Sridhar, Oxford (Higher Education).

MICROPROCESSOR & MICROCONTROLLER

[As per Choice Based Credit System (CBCS) scheme]
(Effective from the academic year 2018-2019)
SEMESTER – IV

Subject Code	18CS/IS43	**CIE Marks**	50
Number of Lecture Hours/Week	04	**SEE Marks**	50
Total Number of Lecture Hours	48	**Exam Hours**	03

CREDITS - 04

Course objectives: This course will enable students to

- To impart basic understanding of the internal organization of 8086/88 Microprocessor.
- To introduce the concepts of interfacing microprocessors with external devices.
- To develop Assembly language programming skills.

Module I	Teaching Hours
The x86 microprocessor: Brief history of the x86 family, Inside the 8088/86,Introduction to assembly programming, Introduction to Program Segments, The Stack, Flag register, x86 Addressing Modes. **Assembly language programming:** Directives & a Sample Program, Assemble, Link & Run a program, More Sample programs, Control Transfer Instructions, Data Types and Data Definition,	10
Module II	
x86: Instructions sets description**, Arithmetic and logic instructions and programs:** Unsigned Addition and Subtraction, Unsigned Multiplication and Division, Logic Instructions,BCD and ASCII conversion, Rotate Instructions. **INT 21H and INT 10H Programming:** Bios INT 10H Programming , DOS Interrupt 21H. 8088/86 Interrupts,x86 PC and Interrupt Assignment.	10
Module III	
Signed Numbers and Strings: Signed number Arithmetic Operations, String operations, **Memory and Memory interfacing: M**emory address decoding, data integrity in RAM and ROM, 16-bit memory interfacing. **8255 I/O programming:** I/O addresses MAP of x86 PC's, programming and interfacing the 8255.	08
Module IV	
Microprocessors versus Microcontrollers, **ARM Embedded Systems :**The RISC design philosophy, The ARM Design Philosophy, Embedded System Hardware, Embedded System Software, **ARM Processor Fundamentals :** Registers , Current Program Status Register , Pipeline, Exceptions, Interrupts, and the Vector Table.	10
Module V	
Introduction to the ARM Instruction Set : Data Processing Instructions ,Branch Instructions, Software Interrupt Instructions, Program Status Register Instructions Coprocessor Instructions, Loading Constants.	10

Course Outcomes

After studying this course, students will be able to:

- **CO 1:** Differentiate between microprocessors and microcontrollers
- **CO 2:** Develop assembly language code to solve problems
- **CO 3:** Interface microprocessors with various external devices.
- **CO 4:** Analyze and compare the features of microprocessors.
- **CO-5:** Demonstrate interrupt routines for interfacing devices□

Question paper pattern:

- The question paper will have ten questions.
- There will be 2 questions from each module.

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

Text Books:

1. Muhammad Ali Mazidi, Janice Gillispie Mazidi, Danny Causey, The x86 PC Assembly Language Design and Interfacing, 5th Edition, Pearson, 2013.
2. **ARM system developers guide**, Andrew N Sloss, Dominic Symes and Chris Wright, Elsevier, Morgan Kaufman publishers, 2008.

Reference Books:

1. Douglas V. Hall: Microprocessors and Interfacing, Revised 2nd Edition, TMH, 2006.
2. Ayala : The 8086 Microprocessor: programming and interfacing - 1st edition, Cengage Learning
3. The Definitive Guide to the ARM Cortex-M3, by Joseph Yiu, 2nd Edition , Newnes, 2009
4. The Insider's Guide to the ARM7 based microcontrollers, Hitex Ltd.,1st edition, 2005
5. ARM System-on-Chip Architecture, Steve Furber, Second Edition, Pearson, 2015
6. Architecture, Programming and Interfacing of Low power Processors- ARM7, Cortex-M and MSP430, Lyla B Das Cengage Learning, 1st Edition.

JAVA PROGRAMMING

[As per Choice Based Credit System (CBCS) scheme]
(Effective from the academic year 2017 -2018)
SEMESTER – IV

Subject Code	18CS/IS44	**CIE Marks**	50
Number of Lecture Hours/Week	04	**SEE Marks**	50
Total Number of Lecture Hours	48	**Exam Hours**	03

CREDITS – 04

Course objectives: This course will enable students to

- Learn fundamental features of object oriented language and JAVA
- Set up Java JDK environment to create, debug and run simple Java programs.
- Learn object oriented concepts using programming examples.
- Study the concepts of importing of packages and exception handling mechanism.
- Discuss the String Handling examples with Object Oriented concepts.

Module I	**Teaching Hours**
An Overview of Java: Object-Oriented Programming, A First Simple Program, A Second Short Program, Two Control Statements, Using Blocks of Code, Lexical Issues, The Java Class Libraries, Data Types, Variables, and Arrays: Java Is a Strongly Typed Language, The Primitive Types, Integers, Floating-Point Types, Characters, Booleans, A Closer Look at Literals, Variables, Type Conversion and Casting, Automatic Type Promotion in Expressions, Arrays, A Few Words About Strings	10
Module II	
Operators: Arithmetic Operators, The Bitwise Operators, Relational Operators, Boolean Logical Operators, The Assignment Operator, The ? Operator, Operator Precedence, Using Parentheses, Control Statements: Java's Selection Statements, Iteration Statements, Jump Statements. Introducing Classes: Class Fundamentals, Declaring Objects, Assigning Object Reference Variables, Introducing Methods, Constructors, The this Keyword, Garbage Collection, The finalize( ) Method, A Stack Class	10
Module III	
A Closer Look at Methods and Classes: Overloading Methods, Using Objects as Parameters, A Closer Look at Argument Passing, Returning Objects, Recursion, Introducing Access Control, Understanding static, Introducing final, Arrays Revisited, Inheritance: Inheritance, Using super, Creating a Multilevel Hierarchy, When Constructors Are Called, Method Overriding, Dynamic Method Dispatch, Using Abstract Classes, Using final with Inheritance, The Object Class.	10
Module IV	
Packages and Interfaces: Packages, Access Protection, Importing Packages, Interfaces, Exception Handling: Exception-Handling Fundamentals, Exception Types, Uncaught Exceptions, Using try and catch, Multiple catch Clauses, Nested try Statements, throw, throws, finally, Java's Built-in Exceptions, Creating Your Own Exception Subclasses, Chained Exceptions, Using Exceptions.	08
Module V	
The Applet Class: Introduction, Two types of Applets; Applet basics; Applet Architecture; An Applet skeleton; Simple Applet display methods; Requesting repainting; Using the Status Window; The HTML APPLET tag; Passing parameters to Applets; getDocumentbase() and getCodebase(); ApletContext and showDocument(); The AudioClip Interface; The AppletStub Interface; Output to the Console. String	10

Handling: The String Constructors, String Length, Special String Operations, Character Extraction, String Comparison, Searching Strings, Modifying a String, Data Conversion Using valueOf( ), Changing the Case of Characters Within a String , Additional String Methods, StringBuffer, StringBuilder.

Course Outcomes

The students shall able to:
CO1: Explain the object-oriented concepts and JAVA.
CO2: Develop computer programs to solve real world problems in Java.
CO3: Develop simple GUI interfaces for a computer program to interact with users

Question paper pattern:
The question paper will have ten questions.
There will be 2 questions from each module.
Each question will have questions covering all the topics under a module.
The students will have to answer 5 full questions, selecting one full question from each module.

Text Books:
1. Herbert Schildt, Java The Complete Reference, 7th Edition, Tata McGraw Hill, 2007. (Chapters 2, 3, 4, 5, 6,7, 8, 9,10, 12,13,15)

Reference Books:
1. Mahesh Bhave and Sunil Patekar, "Programming with Java", First Edition, Pearson Education,2008, ISBN:9788131720806.
2.RajkumarBuyya,SThamarasiselvi, xing chenchu, Object oriented Programming with java, Tata McGraw Hill education private limited.
3. E Balagurusamy, Programming with Java A primer, Tata McGraw Hill companies.
4. Anita Seth and B L Juneja, JAVA One step Ahead, Oxford University Press, 2017.

MICROPROCESSOR AND MICROCONTROLLER LAB

[As per Choice Based Credit System (CBCS) scheme]
(Effective from the academic year 2018-2019)
SEMESTER – IV

Subject Code	18CSL/ISL45	**CIE Marks**	50
Number of Lecture Hours/Week	02	**SEE Marks**	50
Total Number of Lecture Hours	48	**Exam Hours**	03

CREDITS – 01

Course objectives: This course will enable students to

- Demonstration and Explanation of hardware components ,8086 architecture, pin diagram
- Develop and execute the following programs using 8086 Assembly Language. Any suitable assembler like MASM/TASM/8086 kit or any equivalent software may be used.

Laboratory Session-1: Write-up on Microprocessors, 8086 Functional block diagram, Pin diagram and description. The same information is also taught in theory class; this helps the students to understand better

Laboratory Session-2: Write-up on Instruction group, Timing diagrams, etc. The same information is also taught in theory class; this helps the students to understand better.

Note: These TWO Laboratory sessions are used to fill the gap between theory classes and practical sessions. Both sessions are to be evaluated for 20 marks as lab experiments.

PART – A

1. Design and develop an assembly language program to search a key element “X” in a list of ‘n’ 16-bit numbers. Adopt Binary search algorithm in your program for searching.
2. Design and develop an assembly program to sort a given set of ‘n’ 16-bit numbers in ascending order. Adopt Bubble Sort algorithm to sort given elements.
3. Design and develop an assembly language program to read the current time and Date from the system and display it in the standard format on the screen.
4. Develop an assembly language program to reverse a given string and verify whether it is a palindrome or not. Display the appropriate message.
5. Design an assembly language program to compute nCr using recursive procedure. Assume that ‘n’ and ‘r’ are non-negative integers.
6. Design an assembly language program to create a file and delete an existing file.
7. To write and simulate C Program to ARM microprocessor using KEIL. (Demonstrate with the help of suitable program)

PART – B

1. Design and develop an assembly program to read the status of two 8-bit inputs (X & Y) from the Logic Controller Interface and display X*Y.
2. Design and develop BCD Up-Down counter using Logic Controller Interface.
3. Design and develop an assembly program to display messages “FIRE” and “HELP” alternately with flickering effects on a 7-segment display interface for a suitable period of time.
4. To interface stepper motor with ARM processor- ARM7TDMI/LPC2148. Write a program to rotate stepper motor.

Course Outcomes

The students should be able to:
CO 1: Program a microprocessor to perform arithmetic, logical and data transfer applications.
CO 2: Understand assembler directives, DOS Interrupts, branch and loop operations.
CO 3: Interface a microprocessor to various devices for simple applications.
CO 4: Effectively utilize microprocessor peripherals.
CO 5: Utilize procedures and macros for modular programming

JAVA PROGRAMMING LAB

[As per Choice Based Credit System (CBCS) scheme]

(Effective from the academic year 2018-2019)

SEMESTER – IV

Subject Code	18CSL/ISL46	**CIE Marks**	50
Number of Lecture Hours/Week	02	**SEE Marks**	50
Total Number of Lecture Hours	48	**Exam Hours**	03

CREDITS - 01

Course objectives: This course will enable students

- Learn fundamental features of object oriented language and JAVA
- Set up Java JDK environment to create, debug and run simple Java programs.
  Learn object oriented concepts using programming examples

PART – A

1. a. Write a JAVA program to implement class mechanism. –Create a class, methods and invoke them inside main method.
   b. Write a JAVA program to implement shift operators in JAVA
2. a. Write a JAVA program to implement constructor overloading.
   b. Write a JAVA program to implement for-each loop to compute average of n natural numbers.
3. a. Write a JAVA program to implement multi level Inheritance.
   b. Write a JAVA program for abstract class to find areas of different shapes.
4. a. Write a JAVA program that describes exception handling mechanism.
   b. Write a JAVA program to implement break and continue statements.
5. a. Write a JAVA program using IO Streams.
   b. Write a JAVA program using files.

PART – B (Implement the following in JAVA)

1. Write a JAVA program that creates threads by extending Thread class .First thread display “Good Morning “every 1 sec, the second thread displays “Hello “every 2 seconds and the third display “Welcome” every 3 seconds,(Repeat the same by implementing Runnable.
2. Write a JAVA program Producer Consumer Problem.
3. Write a JAVA program to create an applet and set its background color and foreground color displaying a message
4. A. Write a JAVA program to demonstrate key event handlers using delegation event model.

The students should be able to:
CO 1: Implement the java program using constructor, inheritance.
CO2: Implement the java program using exception handling.
CO2: Implement the java program using threads.

DESIGN AND ANALYSIS OF ALGORITHM LAB

[As per Choice Based Credit System (CBCS) scheme]

(Effective from the academic year 2018-2019)

SEMESTER – IV

Subject Code	18CSL/ISL47	**CIE Marks**	50
Number of Lecture Hours/Week	02	**SEE Marks**	50
Total Number of Lecture Hours	48	**Exam Hours**	03

CREDITS - 01

Course objectives: This course will enable students

- Design and implement various algorithms in JAVA
- Employ various design strategies for problem solving.
- Measure and compare the performance of different algorithms.

PART – A

1. Design a program to search a key element of n integers using binary search algorithm and compute time complexity
2. Design a program to Sort a given set of n integer elements using Quick Sort method and compute its time complexity.
3. Design a program to sort set of n integer elements using Merge Sort method and compute its time complexity.
4. Implement the 0/1 Knapsack problem using
   (a) Dynamic Programming method.
   (b) Greedy method.
5. Design a program to print all the node reachable from a given starting node in a given digraph using DFS method.

PART – B (Implement the following in JAVA)

1. Write a Program find shortest paths to other vertices using Dijkstra's algorithm.
2.
   (a) Write a program to find a Minimum Cost Spanning Tree of a given connected undirected graph using Kruskal's algorithm.
   (b) Write a program to find Minimum Cost Spanning Tree of a given connected undirected graph using Prim's algorithm.
3. Write a program to
   (a) Implement All-Pairs Shortest Paths problem using Floyd's algorithm.
   (b) Implement transitive closure using warshall Algorithm.
4. Design and implement to find a subset of a given set.
5. Implement Travelling Salesman problem using Dynamic program.

Course Outcomes

The students should be able to:

CO 1: Design algorithms using appropriate design techniques (brute-force, greedy, dynamic programming, etc.)

CO 2: Develop variety of algorithms such as sorting, graph related, combinatorial, etc., in a high level language.

CO 3: Analyze and compare the performance of algorithms using language features.

CO 4: Apply and implement learned algorithm design techniques and data structures to solve real-world problems. Will be able to do Basic System administration.

ADDITIONAL MATHEMATICS - I

(B.Tech. III semester Common to all branches)

(A Bridge course for Lateral Entry students of III Sem. B. Tech.)

[As per Choice Based Credit System (CBCS) scheme]
(Effective from the academic year 2019-20)

Course Code : 19MATDIP31	**CIE Marks : 00**
Contact Hours/Week : 03	**SEE Marks: 100**
Total Hours:40	**Exam Hours:03**
Semester : III	**Credits: 00**

Course Learning Objectives:

This course will enable students to:

- Acquire basic concepts of complex trigonometry, vector algebra, differential & integral calculus and vector differentiation.
- Evaluation of double and triple integrals.
- know the basic concepts of partial differential equations.
- To develop the knowledge of matrices and linear algebra in compressive manner.
- To understand the essential concept of linear algebra.

MODULE-I

Complex Trigonometry-1:

Complex Numbers: Definition and Properties . Modulus and Amplitude of complex number, Argand's diagram , De-Moivre's theorem (without proof)

Vector Analysis : Scalar and Vectors. Vector addition and subtraction. Multiplication of vectors (Dot and Cross products) Scalar and vector triple products- simple problems, Vector Differentiation : Gradient, Divergence and Curl.

8 - Hours

MODULE-II

Differential Calculus:

Review of successive differentiation. Formulae of Nth derivatives of standard functions- Leibnitz's theorem (without proof).

Polar Curves: Expression for Angle between radius vector and tangent, length of perpendicular from pole to the tangent, angle between two polar curves, Pedal Equation of polar curves and problems. Taylor' and Maclaurin's seires expansions.

8 - Hours

MODULE-III

Partial Differentiation :

Definitions of Partial Differentiation, Direct and Indirect partial derivatives, Symmetric functions, Homogeneous function and Euler's theorem on homogeneous function. Total Derivative of composite and implicit function. Jacobian.

8 - Hours

MODULE-IV

Integral Calculus:

Reduction Formulae of $\int_0^{\pi/2} Sin^n xdx$, $\int_0^{\pi/2} Cos^n xdx$, and Statement of Reduction formulae $\int_0^{\pi/2} Sin^m xCos^n xdx$ and Problems.

Double and Triple integral- simple problems. **8 - Hours**

MODULE-V

Linear Algebra:

Basic concepts of matrices- Rank of matrix by elementary row transformations- Echelon form. Consistency of system of Linear equations. Solution of system linear equations by Gauss Elimination method, Linear Transformation, Cayley- Hamilton theorem to compute inverse of matrix. Eigen values and Eigen vector, Largest eigen values of eigen vectors by Reyleigh's Power method. **8 - Hours**

Course outcomes:

On completion of the course, students are able to:

- Understand the fundamental concepts of complex numbers and vector algebra to analyze the problems arising in related area.
- Use derivatives and partial derivatives to calculate rates of change of multivariate functions.
- Learn techniques of integration including double and triple integrals to find area, volume, mass and moment of inertia of plane and solid region.
- Analyze position, velocity and acceleration in two or three dimensions using the calculus of vector valued functions.
- Recognize and solve first-order ordinary differential equations occurring in different branches of engineering.
- Solve systems of linear equations in the different areas of linear algebra.

Question paper pattern:

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

Text Book:

B.S. Grewal: Higher Engineering Mathematics, Khanna Publishers, New Delhi, 43rd Ed., 2015.

Reference Books:

1. E. Kreyszig: Advanced Engineering Mathematics, John Wiley & Sons, 10th Ed., 2015.

2. N.P.Bali and Manish Goyal: Engineering Mathematics, Laxmi Publishers, 7th Ed., 2007.

ADDITIONAL MATHEMATICS - II

(B.Tech. III semester Common to all branches)

(A Bridge course for Lateral Entry students of IV Sem. B.Tech.)

[As per Choice Based Credit System (CBCS) scheme]
(Effective from the academic year 2019-20)

Course Code : 19MATDIP41 **CIE Marks : 00**
Contact Hours/Week : 03 **SEE Marks:**
100
Total Hours:40 **Exam Hours:03**
Semester : IV **Credits: 00**

Course Learning Objectives:

This course will enable students to:

- Solve first order differential equations. .
- Solve second and higher order differential equations.
- Understand and solve the partial differential equation.
- To acquire the knowledge of elementary probability theory.
- Know the basic concepts of evaluation of double and triple integrals.

MODULE-I

Differential Equation-1:-

Solution of first order and first degree differential equations: Variable separable, Homogeneous, Exact and Reducible to exact differential equation, Linear differential equation. Applications of first order first degree differential equations: Newton’s law of cooling.

8 - Hours

MODULE-II

Differential Equations-2:-Solution of second & higher order Ordinary linear differential equation with constant co-efficients.Method of variation of parameters. Solution of homogeneous LDE by Power series solution Method.

8 - Hours

MODULE-III

Partial Differential Equations(PDE’s):- Formation of PDE by eliminating arbitrary constant & functions, Solution of Non-homogeneous PDE by direct integration, solution of homogeneous PDE with respect to one independent variable only. Derivation of one dimensional wave equation and heat equation and Various possible solution of wave & heat equations by methods of separation of variables.

8 - Hours

MODULE-IV

Improper Integrals: Beta and gamma functions and its properties and examples.
Evaluation of double integral over a specific region, changing the order of integration , changing into polar form.

8 - Hours

MODULE-V

Probability: Introduction , Sample space and Events. Axioms of Probability, Addition & Multiplication theorems. Conditional probability- illustrative examples. Baye's theorem-examples.

8 – Hours

Course Outcomes:

On completion of this course, students are able to:

- Solve first order differential equations in the different areas of Engineering.
- Solve second and higher order differential equations occurring in of electrical circuits, damped/un-damped vibrations.
- Solve second order partial differential equations in the different areas in the real world.
- Recall basic concepts of elementary probability theory and, solve problems related tothe decision theory, synthesis and optimization of digital circuits.
- To find the surface area and volume of 3D objects.

Question paper pattern:

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

Text Book:

B.S. Grewal: Higher Engineering Mathematics, Khanna Publishers, 43rd Ed., 2015.

Reference Books:

1. E. Kreyszig: Advanced Engineering Mathematics, John Wiley & Sons, 10th Ed., 2015.

2. N.P.Bali and Manish Goyal: A Text Book of Engineering Mathematics, Laxmi Publishers,7th Ed., 2007.

DATA BASE MANAGEMENT SYSTEM

[As per Choice Based Credit System (CBCS) scheme]
(Effective from the academic year 2018-2019)
SEMESTER – V

Subject Code	18CS/IS51	**CIE Marks**	50
Number of Lecture Hours/Week	03	**SEE Marks**	50
Total Number of Lecture Hours	48	**Exam Hours**	03

CREDITS – 04

Course objectives: This course will enable students

- To learn the fundamentals of data models and to represent a database system using ER diagrams.
- To study SQL and relational database design.
- To understand the internal storage structures using different file and indexing techniques which will help in physical DB design.
- To understand the fundamental concepts of transaction processing- concurrency control techniques and recovery procedures.
- To have an introductory knowledge about the Storage and Query processing Techniques

Module I	**Teaching Hours**
Introduction to Databases: Introduction, Characteristics of database approach, Advantages of using the DBMS approach. History of database applications. **Overview of Database Languages and Architectures**: Data Models, Schemas, and Instances. Three schema architecture and data independence, database languages, and interfaces. **Conceptual Data Modeling using Entities and Relationships:** Entity types, Entity sets,attributes, roles, and structural constraints, Weak entity types, ER diagrams, examples, Specialization and Generalization. **RBT: L1, L2, L3.**	10Hours
Module II	
Relational Model: Relational Model Concepts, Relational Model Constraints and relational database schemas, Update operations, transactions, and dealing with constraint violations. **Mapping Conceptual Design into a Logical Design**: Relational Database Design using ER-to-Relational mapping. SQL: SQL data definition and data types, specifying constraints in SQL, retrieval queries in SQL, INSERT, DELETE, and UPDATE statements in SQL, Additional features of SQL. **SQL: Advances Queries**: More complex SQL retrieval queries, Specifying constraints as assertions and action triggers, Views in SQL, Schema change statements in SQL. **RBT: L1, L2, L3.**	10 Hours
Module III	
Database Application Development: Accessing databases from applications, An introduction to JDBC, JDBC classes and interfaces, SQLJ, Stored procedures, Case study: The internet Bookshop. **Functional Dependencies and Normalization for Relational Databases**: Informal Design Guidelines for Relation Schemas, Functional Dependencies,	08 Hours

And Normal Forms Based on Primary Keys, General Definitions of Second and Third Normal Forms, Boyce-Code Normal Form. **RBT: L1, L2, L3.**	
Module IV	
Database Design Algorithms: Properties of Relational Decompositions, Algorithms for Relational Database Schema Design, Nulls, Dangling tuples, Further discussion of Multivalued dependencies and 4NF, Other dependencies and Normal Forms **Transaction Management** – Introduction to Transaction processing, Transaction and system concepts, Desirable properties of Transactions, characterizing schedules based on recoverability and Serializability. **RBT: L1, L2, L3.**	10 Hours
Module V	
Concurrency Control in Databases: Two-phase locking techniques for Concurrency control, Concurrency control based on Timestamp ordering, Multiversion Concurrency control techniques, Validation Concurrency control techniques, Granularity of Data items and Multiple Granularity Locking. **Database Recovery Techniques**: Recovery Concepts, Recovery techniques based on Deferred update, Recovery techniques based on immediate update, Shadow paging, Database backup and recovery from catastrophic failures. **RBT: L1, L2, L3.**	10 Hours

Course Outcomes

After studying this course, students will be able to:

CO 1: Identify, analyze and define database objects, enforce integrity constraints on a database using RDBMS.
CO 2: Use Structured Query Language (SQL) for database manipulation.
CO 3: Design and build simple database systems
CO 4: Develop application to interact with databases.

Question paper pattern:

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

Text Books:

1. . Fundamentals of Database Systems, RamezElmasri and Shamkant B. Navathe, 7th Edition, 2017, Pearson.
2. 2. Database management systems, Ramakrishnan, and Gehrke, 3rd Edition, 2014, McGraw Hill

Reference Books:

1. SilberschatzKorth and Sudharshan, Database System Concepts, 6th Edition, McGrawHill, 2013.

2. Coronel, Morris, and Rob, Database Principles Fundamentals of Design, Implementation and Management, Cengage Learning 2012.

COMPUTER NETWORKS

[As per Choice Based Credit System (CBCS) scheme]
(Effective from the academic year 2018-2019)
SEMESTER – V

Subject Code	**18CS/IS52**	**CIE Marks**	**50**
Number Lecture Hour/Week	**03**	**SEE Marks**	**50**
Number of Lecture Hours	**48**	**Exam Hours**	**03**

CREDITS-02

Course objectives:

- Demonstration of application layer protocols.
- Discuss transport layer services and understand UDP and TCP protocols.
- Explain routers, IP and Routing Algorithms in network layer.
- Demonstration of application layer protocols
- Discuss transport layer services and understand UDP and TCP

Modules	**Teaching Hours**
Module -1	
Introduction - Hardware and software, Data communication, Networking, Protocols and Protocol architecture, standards. Data transmission concepts. Analog and digital transmission. Transmission impairments. Layered Architecture of Computer Networks, OSI and TCP/IP architectures	10 Hours
Module -2	
Physical Layer and Data link Layer – 1 - Guided transmission media and wireless transmission, Multiplexing, Spread spectrum. Switching: Introduction, Circuit-Switched networks, packet switching. Data link layer: Introduction, Link-layer addressing. error detection and Correction: Introduction, Block Coding, Cyclic codes, checksum, forward error correction	10 Hours L1,L2
Module -3	
Data **Link Layer-2:** DLC services, Data Link layer protocols, HDLC, PPP, Random access, Controlled access, Channelization, Ethernet protocol, Standard Ethernet, Fast Ethernet.	10 Hours L1,L2
Module-4	
Network and Transport Layer: Network layer services, Packet switching, Network layer performance, IPV4 addresses, Forwarding of IP packets, IP, ICMPv4,Mobile IP. Unicast routing: Introduction, Routing algorithms, Unicast routing protocols	10 Hours L1, L2
Module-5	
Transport Layer and Application Layer: IPv6 addressing, IPv6 protocol, transport layer protocols: Introduction, UDP, TCP, Standard Client-server protocols: WWW and HTTP, FTP, Electronic mail, DN	08 Hours

	L1,L2

Course outcomes:

At the end of the course the student will be able to:

CO 1: Illustrate basic computer network technology.

CO 2: Identify the different types of network topologies and protocols.

CO 3: Enumerate the layers of the OSI model and TCP/IP functions of each layer.

CO 4: Explain principles of application layer protocols

CO 5: Recognize transport layer services and infer UDP and TCP protocols and Classify routers, IP and Routing Algorithms in network layer

Text Books:

1. Data communication & Networks , by Behrouz A. Forouzan, Tata McGraw Hill. 2002 .

Reference Books:

1. Data Communications, Computer networking on OSI, by Fred Halsall, Addison Wesley Publishing Co. 1998.
2. Computer Networking -A Top-Down Approach Featuring the Internet, James F. Kurose and Keith W. Ross , Addison Wesley Publishing Co. 2004
3. Computer Networks: Protocols standards and interfaces, by Uyless Black, Prentice Hall 2002.
4. Computer Networks, by Andrew S. Tanenbaum, PHI. (2010)
5. Data and Computer Communications, by Walliam Stallings, PHI. (2002)

WEB PROGRAMMING LABORATORY

[As per Choice Based Credit System (CBCS) scheme]

(Effective from the academic year 2019-2020)

SEMESTER – V

Subject Code	18CS/IS55	**CIE Marks**	50
Number of Lecture Hours/Week	02	**SEE Marks**	50
Total Number of Lecture Hours	48	**Exam Hours**	03

CREDITS – 02

PART-A

1. Write a JavaScript to design a simple calculator to perform the following operations: sum, product, difference and quotient.

2. Write a JavaScript that calculates the squares and cubes of the numbers from 0 to 10 and outputs HTML text that displays the resulting values in an HTML table format.

3. Write a JavaScript code that displays text “TEXT-GROWING” with increasing font size in the interval of 100ms in RED COLOR, when the font size reaches 50pt it displays “TEXT-SHRINKING” in BLUE color. Then the font size decreases to 5pt.

4. Develop and demonstrate a HTML5 file that includes JavaScript script that uses functions for the following problems:

a. Parameter: A string

b. Output: The position in the string of the left-most vowel

c. Parameter: A number
d. Output: The number with its digits in the reverse order

5. Design an XML document to store information about a student in an engineering college affiliated to VTU. The information must include USN, Name, and Name of the College, Branch, Year of Joining, and email id. Make up sample data for 3 students. Create a CSS style sheet and use it to display the document.

6. Write a PHP program to keep track of the number of visitors visiting the web page and to display this count of visitors, with proper headings.

7. Write a PHP program to display a digital clock which displays the current time of the server.

8. Write the PHP programs to do the following:
a. Implement simple calculator operations.
b. Find the transpose of a matrix.
c. Multiplication of two matrices.
d. Addition of two matrices.

9. Write a PHP program named states.py that declares a variable states with value "Mississippi Alabama Texas Massachusetts Kansas". write a PHP program that does the following:
a. Search for a word in variable states that ends in xas. Store this word in element 0 of a list named statesList.
b. Search for a word in states that begins with k and ends in s. Perform a caseinsensitive comparison. [Note: Passing re.Ias a second parameter to method compile performs a case-insensitive comparison.] Store this word in element1 of statesList.
c. Search for a word in states that begins with M and ends in s. Store this word in element 2 of the list.
d. Search for a word in states that ends in a. Store this word in element 3 of the list.

10. Write a PHP program to sort the student records which are stored in the database using selection sort.

Course Outcomes

- Design and develop dynamic web pages with good aesthetic sense of designingand latest technical know-how's.
- Understand the concepts of Web Application Terminologies, Internet Tools otherweb services.
- Recall how to link and publish web sites

Conduct of Practical Examination:

- Experiment distribution
  - For laboratories having only one part: Students are allowed to pick one experiment from the lot with equal opportunity.
  - For laboratories having PART A and PART B: Students are allowed to pick one experiment from PART A and one experiment from PART B, with equal opportunity.
- Change of experiment is allowed only once and marks allotted for procedure to be made zero of the changed part only.
- Marks Distribution (Courseed to change in accoradance with university regulations)

- For laboratories having only one part – Procedure + Execution + Viva-Voce: 15+30+5 = 50 Marks
- For laboratories having PART A and PART B
  i. Part A – Procedure + Execution + Viva = 6 + 12 + 2 = 20 Marks
  ii. Part B – Procedure + Execution + Viva = 7 + 20 + 3= 30 Marks

DATABASE MANAGEMENT LAB
[As per Choice Based Credit System (CBCS) scheme]
(Effective from the academic year 2019-2020)
SEMESTER – VI

Subject Code	18CSL57	**CIE Marks**	50
Number of Lecture Hours/Week	02	**SEE Marks**	50
Total Number of Lecture Hours	48	**Exam Hours**	03

CREDITS – 02

Objectives:

- Able to understand the Knowledge about basic SQL fundamentals and table operations.
- Able to understand the implementation of Various SQL commands
- Understand the working of Commit and Rollback
- Understand the implementation of nested queries and transactions.

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, IS NULL 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 (Equi-Join, Non-EquiJoin , 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.
11. Create FORMS and REPORTS.
12. Creation, altering and droping of tables and inserting rows into a table (use constraints while creating tables) examples using SELECT command.

Course Outcomes

After studying this course, students will be able to:

- Knowledge about basic SQL fundamentals and table operations.
- Practical implementation of SQL commands and understanding about the working various operators like AND, IN, UNION, INTERSECT etc.
- Knowledge about the working of COMMIT and ROLLBACK
- Implementation of nested queries and handling online transactions.

Conduct of Practical Examination:

- Experiment distribution
  - For laboratories having only one part: Students are allowed to pick one experiment from the lot with equal opportunity.
  - For laboratories having PART A and PART B: Students are allowed to pick one experiment from PART A and one experiment from PART B, with equal opportunity.
- Change of experiment is allowed only once and marks allotted for procedure to be made zero of the changed part only.
- Marks Distribution (Course to change in accordance with university regulations)
- For laboratories having only one part – Procedure + Execution + Viva-Voce: 15+30+5 = 50Marks

Computer Networks Lab

[As per Choice Based Credit System (CBCS) scheme]

(Effective from the academic year 2019-2020)

SEMESTER – V

Subject Code	18CSL58	**CIE Marks**	50
Number of Lecture Hours/Week	02	**SEE Marks**	50
Total Number of Lecture Hours	48	**Exam Hours**	03

CREDITS – 02

PART-A

Objectives:

- To understand the functionalities of various layers of OSI model
- To understand the operating system functionalities

Implement the following using C/C++ or equivalent with LINUX/ Windows environment:

13. Using TCP/IP Socket programming, implement a program to transfer the contents of a requested file from server to the client using TCP/IP Sockets.
14. Implement the data link layer farming methods such as character, character stuffing and bit stuffing.
15. Implement on a data set of characters the three CRC polynomials - CRC 12, CRC 16 and CRC CCIP.
16. Write a program for frame sorting technique used in buffers.
17. Write a program for Hamming Code generation for error detection and correction.
18. Take an example subnet graph with weights indicating delay between nodes. Now obtain Routing table at each node using distance vector routing algorithm.
19. Using Leaky Bucket Algorithm, Design a program to achieve Traffic management at Flow level by implementing Closed Loop Control technique.
20. Using RSA algorithm encrypt a text data and Decrypt the same.

Part B

Simulation Programs using any network simulator or any other equivalent software.
Note: (i) Analyze the network behavior by collecting the statistics on network performance and draw the conclusion.

1. Simulate a 3 node point to point network with duplex links between them. Set the Queue size and vary the bandwidth and find the number of packets dropped.
2. Simulate a four-node point-to-point network, and connect the links as follows: n0->n2, n1- >n2 and n2-n3. Apply TCP agent changing the parameters and determine the number of packets sent/received by TCP/UDP.
3. Implement an Ethernet LAN using n nodes and set multiple traffic nodes and plot congestion window for different source / destination.
4. Implement simple ESS and with transmitting nodes in wire-less LAN by simulation and determine the performance with respect to transmission of packets.
5. Implement and study the performance of GSM on NS2/NS3 (Using MAC layer) or equivalent environment.
6. Implement and study the performance of CDMA on NS2/NS3 (Using stack called Call net) or equivalent environment.

Course Outcomes

After studying this course, students will be able to:

- Analyze and Compare various networking protocols.
- Demonstrate the working of different concepts of networking.
- Ability to understand the encryption and decryption concepts.
- Ability to apply appropriate algorithm for the finding of shortest route.
- Ability to configure the routing table.
- Implement and analyze networking protocols in NS2 / NS3.

Conduct of Practical Examination:

- Experiment distribution
  - For laboratories having only one part: Students are allowed to pick one experiment from the lot with equal opportunity.
  - For laboratories having PART A and PART B: Students are allowed to pick one experiment from PART A and one experiment from PART B, with equal opportunity.
- Change of experiment is allowed only once and marks allotted for procedure to be made zero of the changed part only.
- Marks Distribution (Course to change in accordance with university regulations)
- For laboratories having only one part – Procedure + Execution + Viva-Voce: 15+30+5 = 50Marks
- For laboratories having PART A and PART B

  iii. Part A – Procedure + Execution + Viva = 7 + 20 + 3 = 20 Marks

  iv. Part B – Procedure + Execution + Viva = 6 + 12 + 2= 30 Marks

AUTOMATA THEORY AND COMPUTABILITY

[As per Choice Based Credit System (CBCS) scheme]
(Effective from the academic year 2018-2019)
SEMESTER – V

Subject Code	**18CS531/IS531**	**CIE Marks**	**50**
Number Lecture Hour/Week	**03**	**SEE Marks**	**50**
Number of Lecture Hours	**48**	**Exam Hours**	**03**

CREDITS-04

Course objectives:

This course will enable students to:

- Introduce core concepts in Automata and Theory of Computation
- Identify different Formal language Classes and their Relationships
- Design Grammars and Recognizers for different formal languages
- Prove or disprove theorems in automata theory using their properties
- Determine the decidability and intractability of Computational problems

Modules	**Teaching Hours**
Module -1	
Why study the Theory of Computation, Languages and Strings: Strings, Languages. A Language Hierarchy, Computation, **Finite State Machines (FSM)**: Deterministic FSM, Regular languages, Designing FSM, Nondeterministic FSMs, From FSMs to Operational Systems, Simulators for FSMs, Minimizing FSMs. **Textbook 1: Ch 1,2, 3,4, 5.1 to 5.10** **RBT: L1, L2**	10 Hours
Module -2	
Regular Expressions (RE): what is a RE?, Kleene’s theorem, Applications of REs, Manipulating and Simplifying REs. Regular Grammars: Definition, Regular Grammars and Regular languages. Regular Languages (RL) and Non-regular Languages: How many RLs, To show that a language is regular, Closure properties of RLs, to show some languages are not RLs. **Textbook 1: Ch 6, 7, 8: 6.1 to 6.4, 7.1, 7.2, 8.1 to 8.4** **RBT: L1, L2, L3**	10 Hours
Module -3	
Context-Free Grammars(CFG): Introduction to Rewrite Systems and Grammars, CFGs and languages, designing CFGs, simplifying CFGs, proving that a Grammar is correct, Derivation and Parse trees, Ambiguity, Normal Forms. Pushdown Automata (PDA): Definition of non-deterministic PDA, Deterministic and Non-deterministic PDAs, Nondeterminism and Halting, alternative equivalent definitions of a PDA, alternatives that are not equivalent to PDA. **Textbook 1: Ch 11, 12: 11.1 to 11.8, 12.1, 12.2, 12,4, 12.5, 12.6** **RBT: L1, L2, L3**	10 Hours
Module-4	
Algorithms and Decision Procedures for CFLs: Decidable questions, Un-decidable questions. **Turing Machine**: Turing machine model, Representation, Language acceptability	10 Hours

by TM, design of TM, Techniques for TM construction. Variants of Turing Machines (TM), The model of Linear Bounded automata. **Textbook 1: Ch 14: 14.1, 14.2, Textbook 2: Ch 9.1 to 9.8** **RBT: L1, L2, L3**	
Module-5	
Decidability: Definition of an algorithm, decidability, decidable languages, Undecidable languages, halting problem of TM, Post correspondence problem. Complexity: Growth rate of functions, the classes of P and NP, Quantum Computation: quantum computers, Church-Turing thesis. **Applications:** G.1 Defining syntax of programming language, Appendix J: Security **Textbook 2: 10.1 to 10.7, 12.1, 12.2, 12.8, 12.8.1, 12.8.2** **Textbook 1: Appendix: G.1(only), J.1 & J.2** **RBT: L1, L2, L3**	08 Hours

Course outcomes:

At the end of the course the student will be able to:

- Acquire fundamental understanding of the core concepts in automata theory and Theory of Computation
- Learn how to translate between different models of Computation (e.g., Deterministic and Non-deterministic and Software models).
- Design Grammars and Automata (recognizers) for different language classes and become knowledgeable about restricted models of Computation (Regular, Context Free) and their relative powers.
- Develop skills in formal reasoning and reduction of a problem to a formal model, with an emphasis on semantic precision and conciseness.
- Classify a problem with respect to different models of Computation.

Textbooks:

1. Elaine Rich, Automata, Computability and Complexity, 1st Edition, Pearson education,2012/2013.
2. K L P Mishra, N Chandrasekaran , 3rd Edition, Theory of Computer Science, PhI, 2012.

Reference Books:

1. John E Hopcroft, Rajeev Motwani, Jeffery D Ullman, Introduction to Automata Theory, Languages, and Computation, 3rd Edition, Pearson Education, 2013
2. Michael Sipser : Introduction to the Theory of Computation, 3rd edition, Cengage learning,2013
3. John C Martin, Introduction to Languages and The Theory of Computation, 3rd Edition, Tata McGraw –Hill Publishing Company Limited, 2013
4. Peter Linz, “An Introduction to Formal Languages and Automata”, 3rd Edition, Narosa Publishers, 1998
5. Basavaraj S. Anami, Karibasappa K G, Formal Languages and Automata theory, Wiley India, 2012
6. C K Nagpal, Formal Languages and Automata Theory, Oxford University press, 2012.

CLOUD COMPUTING AND ITS APPLICATIONS

[As per Choice Based Credit System (CBCS) scheme]
(Effective from the academic year 2018-2019)
SEMESTER – V

Subject Code	**18CS532/IS532**	**CIE Marks**	**50**
Number Lecture Hour/Week	**03**	**SEE Marks**	**50**
Number of Lecture Hours	**48**	**Exam Hours**	**03**

CREDITS-03

Course objectives:

- Explain the fundamentals of cloud computing
- Illustrate the cloud application programming and aneka platform
- Contrast different cloud platforms used in industry

Modules	**Teaching Hours**
Module -1	
Introduction ,Cloud Computing at a Glance, The Vision of Cloud Computing, Defining a Cloud, A Closer Look, Cloud Computing Reference Model, Characteristics and Benefits, Challenges Ahead, Historical Developments, Distributed Systems, Virtualization, Web 2.0, Service-Oriented Computing, Utility-Oriented Computing, Building Cloud Computing Environments, Application Development, Infrastructure and System Development, Computing Platforms and Technologies, Amazon Web Services (AWS), Google App Engine, Microsoft Azure, Hadoop, Force.com and Salesforce.com, Manjrasoft Aneka Virtualization, Introduction, Characteristics of Virtualized, Environments Taxonomy of Virtualization Techniques, Execution Virtualization, Other Types of Virtualization, Virtualization and Cloud Computing, Pros and Cons of Virtualization, Technology Examples Xen: Paravirtualization, VMware: Full Virtualization, Microsoft Hyper-V **Textbook 1: Ch. 1,3 RBT: L1, L2**	10 Hours
Module -2	
Cloud Computing Architecture, Introduction, Cloud Reference Model, Architecture, Infrastructure / Hardware as a Service, Platform as a Service, Software as a Service, Types of Clouds, Public Clouds, Private Clouds, Hybrid Clouds, Community Clouds, Economics of the Cloud, Open Challenges, Cloud Definition, Cloud Interoperability and Standards Scalability and Fault Tolerance Security, Trust, and Privacy Organizational Aspects Aneka: Cloud Application Platform, Framework Overview, Anatomy of the Aneka Container, From the Ground Up: Platform Abstraction Layer, Fabric Services, foundation Services, Application Services, Building Aneka Clouds, Infrastructure Organization, Logical Organization, Private Cloud Deployment Mode, Public Cloud Deployment Mode, Hybrid Cloud Deployment Mode, Cloud Programming and Management, Aneka SDK, Management Tools **Textbook 1: Ch. 4,5 RBT: L1, L2**	10 Hours
Module -3	
Concurrent Computing: Thread Programming, Introducing Parallelism for Single Machine	

Computation, Programming Applications with Threads, What is a Thread?, Thread APIs, Techniques for Parallel Computation with Threads, Multithreading with Aneka, Introducing the Thread Programming Model, Aneka Thread vs. Common Threads, Programming Applications with Aneka Threads, Aneka Threads Application Model, Domain Decomposition: Matrix Multiplication, Functional Decomposition: Sine, Cosine, and Tangent. High-Throughput Computing: Task Programming, Task Computing, Characterizing a Task, Computing Categories, Frameworks for Task Computing, Task-based Application Models, Embarrassingly Parallel Applications, Parameter Sweep Applications, MPI Applications, Computing Categories, Frameworks for Task Computing, Task-based Application Models, Embarrassingly Parallel Applications, Parameter Sweep Applications, MPI Applications, Workflow Applications with Task Dependencies, Aneka Task-Based Programming, Task Programming Model, Developing Applications with the Task Model, Developing Parameter Sweep Application, Managing Workflows. **Textbook 1: Ch. 6, 7 RBT: L1,L2**	10 Hours
Module-4	
Data Intensive Computing: Map-Reduce Programming, What is Data-Intensive Computing?, Characterizing Data-Intensive Computations, Challenges Ahead, Historical Perspective, Technologies for Data-Intensive Computing, Storage Systems, Programming Platforms, Aneka MapReduce Programming, Introducing the MapReduce Programming Model, Example Application **Textbook 1: Ch. 8 RBT: L1, L2**	10 Hours
Module-5	
Cloud Platforms in Industry, Amazon Web Services, Compute Services, Storage Services, Communication Services, Additional Services, Google App Engine, Architecture and Core Concepts, Application Life-Cycle, Cost Model, Observations, Microsoft Azure, Azure Core Concepts, SQL Azure, Windows Azure Platform Appliance. Cloud Applications Scientific Applications, Healthcare: ECG Analysis in the Cloud, Biology: Protein Structure Prediction, Biology: Gene Expression Data Analysis for Cancer Diagnosis, Geoscience: Satellite Image Processing, Business and Consumer Applications, CRM and ERP, Productivity, Social Networking, Media Applications, Multiplayer Online Gaming. **Textbook 1: Ch. 9,10 RBT: L1, L2**	08 Hours
Course outcomes: The student will be able to: • Explain cloud computing, virtualization and classify services of cloud computing • Illustrate architecture and programming in cloud • Describe the platforms for development of cloud applications and List the application of cloud.	
Textbooks: 1. RajkumarBuyya, Christian Vecchiola, and ThamaraiSelvi Mastering Cloud. Computing McGraw Hill Education **Reference Book:** 1. Dan C. Marinescu, Cloud Computing Theory and Practice, Morgan Kaufmann, Elsevier 2013.	

UNIX PROGRAMMING

[As per Choice Based Credit System (CBCS) scheme]
(Effective from the academic year 2018-2019)
SEMESTER – V

Subject Code	**18CS533/IS533**	**CIE Marks**	**50**
Number Lecture Hour/Week	**03**	**SEE Marks**	**50**
Number of Lecture Hours	**48**	**Exam Hours**	**03**

CREDITS-03

Course objectives:

This course will enable students to:

- Interpret the features of UNIX and basic commands.
- Demonstrate different UNIX files and permissions
- Implement shell programs.
- Explain UNIX process, IPC and signals.

Modules	**Teaching Hours**
Module -1	
Introduction: Unix Components/Architecture. Features of Unix. The UNIX Environment and UNIX Structure, Posix and Single Unix specification. General features of Unix commands/ command structure. Command arguments and options. Basic Unix commands such as echo, printf, ls, who, date,passwd, cal, Combining commands. Meaning of Internal and external commands. The type command: knowing the type of a command and locating it. The root login. Becoming the super user: su command. **Unix files:** Naming files. Basic file types/categories. Organization of files. Hidden files. Standard directories. Parent child relationship. The home directory and the HOME variable. Reaching required files- the PATH variable, manipulating the PATH, Relative and absolute pathnames. Directory commands – pwd, cd, mkdir, rmdir commands. The dot (.) and double dots (..) notations to represent present and parent directories and their usage in relative path names. File related commands – cat, mv, rm, cp, wc and od commands. **RBT: L1, L2**	10 Hours
Module -2	
File attributes and permissions: The ls command with options. Changing file permissions: the relative and absolute permissions changing methods. Recursively changing file permissions. Directory permissions. The shells interpretive cycle: Wild cards. Removing the special meanings of wild cards. Three standard files and redirection. Connecting commands: Pipe. Basic and Extended regular expressions. The grep, egrep. Typical examples involving different regular expressions. Shell programming: Ordinary and environment variables. The .profile. Read and readonly commands. Command line arguments. exit and exit status of a command. Logical operators for conditional execution. The test command and its shortcut. The if, while, for and case control statements. The set and shift commands and handling positional parameters. The here (<<) document and trap command. Simple shell program examples. **RBT: L1, L2**	10 Hours
Module -3	

UNIX File APIs: General File APIs, File and Record Locking, Directory File APIs, Device File APIs, FIFO File APIs, Symbolic Link File APIs. **UNIX Processes and Process Control:** **The Environment of a UNIX Process:** Introduction, main function, Process Termination, Command-Line Arguments, Environment List, Memory Layout of a C Program, Shared Libraries, Memory Allocation, Environment Variables, setjmp and longjmp Functions, getrlimit, setrlimit Functions, UNIX Kernel Support for Processes. **Process Control:** Introduction, Process Identifiers, fork, vfork, exit, wait, waitpid, wait3, wait4 Functions, Race Conditions, exec Functions **RBT: L1, L2, L3**	10 Hours
Module-4	
Changing User IDs and Group IDs, Interpreter Files, system Function, Process Accounting, User Identification, Process Times, I/O Redirection. **Overview of IPC Methods**, Pipes, popen, pclose Functions, Coprocesses, FIFOs, System V IPC, Message Queues, Semaphores. **Shared Memory**, Client-Server Properties, Stream Pipes, Passing File Descriptors, An Open Server-Version 1, Client-Server Connection Functions. **RBT: L1, L2, L3**	10 Hours
Module-5	
Signals and Daemon Processes: Signals: The UNIX Kernel Support for Signals, signal, Signal Mask, sigaction, The SIGCHLD Signal and the waitpid Function, The sigsetjmp and siglongjmp Functions, Kill, Alarm, Interval Timers, POSIX.lb Timers. Daemon Processes: Introduction, Daemon Characteristics, Coding Rules, Error Logging, Client-Server Model. **RBT: L1, L2, L3**	08 Hours

Course outcomes:

At the end of the course the student will be able to:

- Explain Unix Architecture, File system and use of Basic Commands
- Illustrate Shell Programming and to write Shell Scripts
- Categorize, compare and make use of Unix System Calls
- Build an application/service over a Unix system.

Textbooks:

3. Sumitabha Das., Unix Concepts and Applications., 4thEdition., Tata McGraw Hill (Chapter 1,2 ,3,4,5,6,8,13,14)
4. W. Richard Stevens: Advanced Programming in the UNIX Environment, 2nd Edition, Pearson Education, 2005 (Chapter 3,7,8,10,13,15)
5. Unix System Programming Using C++ - Terrence Chan, PHI, 1999. (Chapter 7,8,9,10).

Reference Books:

7. M.G. Venkatesh Murthy: UNIX & Shell Programming, Pearson Education.
8. Richard Blum , Christine Bresnahan : Linux Command Line and Shell Scripting Bible, 2ndEdition, Wiley,2014.

SOCIAL NETWORK ANALYSIS

[As per Choice Based Credit System (CBCS) scheme]
(Effective from the academic year 2018-2019)
SEMESTER – V

Subject Code	18CS534/IS534	**CIE Marks**	50
Number of Lecture Hours/Week	03	**SEE Marks**	50
Total Number of Lecture Hours	50	**Exam Hours**	03

CREDITS – 03

Course objectives: This course will enable students

Discuss essential knowledge of network analysis applicable to real world data, with examples from today’s most popular social networks.

Module I	**Teaching Hours**
Introduction to social network analysis and Descriptive network analysis: Introduction to new science of networks. Networks examples. Graph theory basics. Statistical network properties. Degree distribution, clustering coefficient. Frequent patterns. Network motifs. Cliques and k-cores. **RBT: L1, L2, L3.**	10Hours
Module II	
Network structure, Node centralities and ranking on network: Nodes and edges, network diameter and average path length. Node centrality metrics: degree, closeness and betweenness centrality. Eigenvector centrality and PageRank. Algorithm HITS. **RBT: L1, L2, L3.**	10 Hours
Module III	
Network communities and Affiliation networks: Networks communities. Graph partitioning and cut metrics. Edge betweenness. Modularity clustering. Affiliation network and bipartite graphs. 1-mode projections. Recommendation systems. **RBT: L1, L2, L3.**	08 Hours
Module IV	
Information and influence propagation on networks and Network visualization: Social Diffusion. Basic cascade model. Influence maximization.Most influential nodes in network. Network visualization and graph layouts. Graph sampling. Low -dimensional projections	10 Hours
Module V	
Social media mining and SNA in real world: FB/VK and Twitter analysis: Natural language processing and sentiment mining. Properties of large social networks: friends, connections, likes, re-tweets. **RBT: L1, L2, L3.**	10 Hours

Course Outcomes

After studying this course, students will be able to:

CO 1: Define notation and terminology used in network science.

CO 2: Demonstrate, summarize and compare networks.

CO 3: Explain basic principles behind network analysis algorithms.

CO 4: Analyze real world network
CO5:

Question paper pattern:

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

Text Books/Reference Books:

1. David Easley and John Kleinberg. "Networks, Crowds, and Markets: ReasoningAbout a Highly Connected World." Cambridge University Press 2010.
2. Eric Kolaczyk, Gabor Csardi. “Statistical Analysis of Network Data with R (UseR!)”.Springer, 2014.
3. Stanley Wasserman and Katherine Faust. "Social Network Analysis. Methods and Applications." Cambridge University Press, 1994.

DISCRETE MATHEMATICAL STRUCTURES AND GRAPH THEORY

[As per Choice Based Credit System (CBCS) scheme]
(Effective from the academic year 2018-2019)
SEMESTER – V

Subject Code	**18CS541/IS541**	**CIE Marks**	**50**
Number Lecture Hour/Week	**03**	**SEE Marks**	**50**
Number of Lecture Hours	**48**	**Exam Hours**	**03**

CREDITS-04

Course objectives:
This course will enable students to:

- Provide theoretical foundations of computer science to perceive other courses in the programme.
- Illustrate applications of discrete structures: logic, relations, functions, set theory and counting.
- Describe different mathematical proof techniques,
- Illustrate the importance of graph theory in computer science

Modules	Teaching Hours
Module -1	
Fundamentals of Logic: Basic Connectives and Truth Tables, Logic Equivalence – The Laws of Logic, Logical Implication – Rules of Inference. Fundamentals of Logic contd.: The Use of Quantifiers, Quantifiers, Definitions and the Proofs of Theorems. **Text book 1: Chapter2** **RBT: L1, L2, L3**	10 Hours
Module -2	
Properties of the Integers: The Well Ordering Principle – Mathematical Induction, Fundamental Principles of Counting: The Rules of Sum and Product, Permutations, **Combinations –** The Binomial Theorem, Combinations with Repetition. **Text book 1: Chapter4 – 4.1, Chapter1** **RBT: L1, L2, L3**	10 Hours
Module -3	
Relations and Functions: Cartesian Products and Relations, Functions – Plain and One-to-One, Onto Functions. The Pigeon-hole Principle, Function Composition and Inverse Functions. **Relations:** Properties of Relations, Computer Recognition – Zero-One Matrices and Directed Graphs, Partial Orders – Hasse Diagrams, Equivalence Relations and Partitions. **Text book 1: Chapter5 , Chapter7 – 7.1 to 7.4** **RBT: L1, L2, L3**	10 Hours
Module-4	
The Principle of Inclusion and Exclusion: The Principle of Inclusion and Exclusion, Generalizations of the Principle, Derangements – Nothing is in its Right Place, Rook Polynomials. **Recurrence Relations:** First Order Linear Recurrence Relation, The Second Order Linear Homogeneous Recurrence Relation with Constant Coefficients. **Text book 1: Chapter8 – 8.1 to 8.4, Chapter10 – 10.1, 10.2** **RBT: L1, L2, L3**	10 Hours
Module-5	

Introduction to Graph Theory: Definitions and Examples, Sub graphs, Complements, and Graph Isomorphism, **Trees**: Definitions, Properties, and Examples, Routed Trees, Trees and Sorting, Weighted Trees and Prefix Codes **Text book 1: Chapter11 – 11.1 to 11.2 Chapter12 – 12.1 to 12.4** **RBT: L1, L2, L3**	08 Hours

Course outcomes:

At the end of the course the student will be able to:

- Use propositional and predicate logic in knowledge representation and truth verification.
- Demonstrate the application of discrete structures in different fields of computer science.
- Solve problems using recurrence relations and generating functions.
- Application of different mathematical proofs techniques in proving theorems in the courses.
- Compare graphs, trees and their applications.

Textbooks:

1. Ralph P. Grimaldi: Discrete and Combinatorial Mathematics, 5th Edition, Pearson Education.2004.

Reference Books:

1. Basavaraj S Anami and Venakanna S Madalli: Discrete Mathematics – A Concept based approach, Universities Press, 2016
2. Kenneth H. Rosen: Discrete Mathematics and its Applications, 6th Edition, McGraw Hill, 2007.
3. Jayant Ganguly: A Treatise on Discrete Mathematical Structures, Sanguine-Pearson, 2010.
4. D.S. Malik and M.K. Sen: Discrete Mathematical Structures: Theory and Applications, Thomson, 2004.
5. Thomas Koshy: Discrete Mathematics with Applications, Elsevier, 2005, Reprint 2008.

MOBILE APPLICATION DEVELOPMENT

[As per Choice Based Credit System (CBCS) scheme]
(Effective from the academic year 2018-2019)
SEMESTER – V

Subject Code	**18CS543/IS543**	**CIE Marks**	**50**
Number Lecture Hour/Week	**03**	**SEE Marks**	**50**
Number of Lecture Hours	**48**	**Exam Hours**	**03**

CREDITS-03

Course objectives:

This course will enable students to:

- Learn to setup Android application development environment
- Illustrate user interfaces for interacting with apps and triggering actions
- Interpret tasks used in handling multiple activities
- Identify options to save persistent application data
- Appraise the role of security and performance in Android applications• Create, test and debug Android application by setting up Android development environment

Modules	**Teaching Hours**
Module -1	
Get started, Build your first app, Activities, Testing, debugging and using support libraries **Textbook 1: Lesson 1,2,3** **RBT: L1, L2**	10 Hours
Module -2	
User Interaction, Delightful user experience, Testing your UI **Textbook 1: Lesson 4,5,6** **RBT: L1, L2**	10 Hours
Module -3	
Background Tasks, Triggering, scheduling and optimizing background tasks **Textbook 1: Lesson 7,8** **RBT: L1, L2**	10 Hours
Module-4	
All about data, Preferences and Settings, Storing data using SQLite, Sharing data with content providers, Loading data using Loaders **Textbook 1: Lesson 9,10,11,12** **RBT: L1, L2**	10 Hours
Module-5	
Permissions, Performance and Security, Firebase and AdMob, Publish **Textbook 1: Lesson 13,14,15** **RBT: L1, L2**	08 Hours

Course outcomes:

At the end of the course the student will be able to:

- Create, test and debug Android application by setting up Android development environment
- Implement adaptive, responsive user interfaces that work across a wide range of devices.
- Infer long running tasks and background work in Android applications

- Demonstrate methods in storing, sharing and retrieving data in Android applications
- Analyze performance of android applications and understand the role of permissions and security
- Describe the steps involved in publishing Android application to share with the world

Textbooks:

1. Google Developer Training, "Android Developer Fundamentals Course – Concept Reference", Google Developer Training Team, 2017. https://www.gitbook.com/book/googledeveloper-training/android-developer-fundamentals-course-concepts/details (Download pdf file from the above link)

Reference Books:

1. Erik Hellman, "Android Programming – Pushing the Limits", 1st Edition, Wiley India Pvt Ltd, 2014.
2. Dawn Griffiths and David Griffiths, "Head First Android Development", 1st Edition, O'Reilly SPD Publishers, 2015.
3. J F DiMarzio, "Beginning Android Programming with Android Studio", 4th Edition, Wiley India Pvt Ltd, 2016. ISBN-13: 978-8126565580
4. Anubhav Pradhan, Anil V Deshpande, " Composing Mobile Apps" using Android, Wiley 2014, ISBN: 978-81-265-4660-2

Green Communications
[As per Choice Based Credit System (CBCS) scheme]
(Effective from the academic year 2018-2019)
SEMESTER – V

Subject Code	18CS534/IS534	**CIE Marks**	50
Number of Lecture Hours/Week	03	**SEE Marks**	50
Total Number of Lecture Hours	50	**Exam Hours**	03

CREDITS – 03

Course objectives: This course will enable students to

Module I	**Teaching Hours**
Introduction, Grid Computing Organizations and Their Roles: Early Grid Activities, Current Grid Activities, An Overview of Grid Business Areas, Grid Applications, Grid Infrastructure. Organizations Developing Grid Standards and Best Practice Guidelines, Organizations Developing Grid Computing Toolkits and the Framework, Organizations Building and Using Grid-Based Solutions to Solve Computing, Data and Network Requirements, Commercial Organizations Building and Using Grid-Based Solutions. **RBT: L1, L2, L3.**	10Hours
Module II	
The Grid Computing Anatomy, Road Map: The Grid Problem. Anatomy Computing, Business on Demand and Infrastructure Virtualization, Service-Oriented Architecture and Grid, Semantic Grids**. RBT: L1, L2, L3.**	10 Hours
Module III	
Architectures : Service-Oriented Architecture, Web Services Architecture, XML, Related Technologies and Their Relevance to Web Services, XML Messages and Enveloping, Service Message Description Mechanisms. Relationship between Web Service and Grid Service, Web Service Interoperability and the Role of the WS-I Organization, OGSA Architecture and Goals, Commercial Data Center (CDC), National Fusion Collaboratory (NFS), Online Media and Entrainment.**RBT: L1, L2, L3.**	08 Hours
Module IV	
The OGSA Platform Components, OGSI: Native Platform Services and Transport Mechanisms, OGSA Hosting Environment, Core Networking Services Transport and Security, OGSA Infrastructure, OGSA Basic Services. Grid Services, A High-Level Introduction to OGSI (Open Grid Services Infrastructure). Technical Details of OGSI Specification, Introduction to Service Data Concepts, Grid Service: Naming and Change Management Recommendations. **RBT: L1, L2, L3.**	10 Hours
Module V	
OGSA Basic Services and Toolkit: Common Management Model (CMM), Service Domains, Policy Architecture, Security Architecture, Metering and Accounting. Common Distributed Logging, Distributed Data Access and Replication. GLOBUS GT3 Toolkit Architecture. **RBT: L1, L2, L3.**	10 Hours

Course Outcomes

After studying this course, students will be able to:

Question paper pattern:

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

Text Books/Reference Books:

1. Joshy Joseph, Craig Fellenstein: Grid Computing, IBM Press, 2007.
2. Prabhu: Grid and Cluster Computing, Prentice-Hall of India, 2007.

SYSTEM SOFTWARE AND COMPILERS

[As per Choice Based Credit System (CBCS) scheme]
(Effective from the academic year 2018-2019)
SEMESTER – V

Subject Code	**18CS61**	**CIE Marks**	**50**
Number Lecture Hour/Week	**03**	**SEE Marks**	**50**
Number of Lecture Hours	**48**	**Exam Hours**	**03**

CREDITS-04

Course objectives:

This course will enable students to:

- Define System Software.
- Familiarize with source file, object file and executable file structures and libraries
- Describe the front-end and back-end phases of compiler and their importance to students

Modules	**Teaching Hours**
Module -1	
Introduction to System Software, Machine Architecture of SIC and SIC/XE. **Assemblers:** Basic assembler functions, machine dependent assembler features, machine independent assembler features, assembler design options. Basic Loader Functions **Text book 1: Chapter 1: 1.1,1.2,1.3.1,1.3.2, Chapter2 : 2.1 to 2.4, Chapter 3 ,3.1** **RBT: L1, L2, L3**	10 Hours
Module -2	
Introduction: Language Processors, The structure of a compiler, The evaluation of programming languages, The science of building compiler, Applications of compiler technology. **Lexical Analysis:** The role of lexical analyzer, Input buffering, Specifications of token, recognition of tokens. **Text book 2:Chapter 1 1.1-1.5 Chapter 3: 3.1 – 3.4** **RBT: L1, L2, L3**	10 Hours
Module -3	
Syntax Analysis: Introduction, Context Free Grammars, Writing a grammar, Top Down Parsers, Bottom-Up Parsers **Text book 2: Chapter 4 4.1, 4.2 4.3 4.4 4.5** **RBT: L1, L2, L3**	10 Hours
Module-4	
Lex and Yacc –The Simplest Lex Program, Grammars, Parser-Lexer Communication, A YACC Parser, The Rules Section, Running LEX and YACC, LEX and Hand- Written	10 Hours

Lexers, Using LEX - Regular Expression, Examples of Regular Expressions, A Word Counting Program, Using YACC – Grammars, Recursive Rules, Shift/Reduce Parsing, What YACC Cannot Parse, A YACC Parser - The Definition Section, The Rules Section, The LEXER, Compiling and Running a Simple Parser, Arithmetic Expressions and Ambiguity. **Text book 3: Chapter 1,2 and 3.** **RBT: L1, L2, L3**	
Module-5	
Syntax Directed Translation, Intermediate code generation, Code generation **Text book 2: Chapter 5.1, 5.2, 5.3, 6.1, 6.2, 8.1, 8.2** **RBT: L1, L2, L3**	08 Hours

Course outcomes:

At the end of the course the student will be able to:

- Explain system software
- Design and develop lexical analyzers, parsers and code generators
- Utilize lex and yacc tools for implementing different concepts of system software

Textbooks:

1. 1. System Software by Leland. L. Beck, D Manjula, 3rd edition, 2012
2. 2. Alfred V Aho, Monica S. Lam, Ravi Sethi, Jeffrey D. Ullman , Compilers-Principles, Techniques and Tools, Pearson, 2nd edition, 2007
3. Doug Brown, John Levine, Tony Mason, lex & yacc, O'Reilly Media, October 2012.

Reference Books:

1. Systems programming – Srimanta Pal , Oxford university press, 2016
2. System programming and Compiler Design, K C Louden, Cengage Learning
3. System software and operating system by D. M. Dhamdhere TMG
4. Compiler Design, K Muneeswaran, Oxford University Press 2013.

SYSTEM SOFTWARE and Compiler Design Lab

[As per Choice Based Credit System (CBCS) scheme]

(Effective from the academic year 2019-2020)
SEMESTER – IV

Subject Code	18CSL65	**CIE Marks**	50
Number of Lecture Hours/Week	03	**SEE Marks**	50
Total Number of Lecture Hours	48	**Exam Hours**	03

CREDITS – 02

Course objectives: This course will enable students

- To make students familiar with Lexical Analysis and Syntax Analysis phases of Compiler Design and implement programs on these phases using LEX & YACC tools and/or C/C++/Java
- To enable students to learn different types of CPU scheduling algorithms used in operating system.
- To make students able to implement memory management - page replacement and deadlock handling algorithms

PART A

Execute the following programs using LEX:

1.
   a. Program to count the number of characters, words, spaces and lines in a given input file.
   b. Program to count the numbers of comment lines in a given C program. Also eliminate them and copy the resulting program into separate file.
2.
   a. Program to recognize a valid arithmetic expression and to recognize the identifiers and operators present. Print them separately.
   b. Program to recognize whether a given sentence is simple or compound.
3. Program to recognize and count the number of identifiers in a given input file.

Execute the following programs using YACC:

4.
   a. Program to recognize a valid arithmetic expression that uses operators +, -, * and /.
   b. Program to recognize a valid variable, which starts with a letter, followed by any number of letters or digits.

5.
    a. Program to evaluate an arithmetic expression involving operators +, -, * and /.
    b. Program to recognize strings ‘aaab’, ‘abbb’, ‘ab’ and ‘a’ using the grammar (anbn, n>= 0).
6. Program to recognize the grammar (anb, n>= 10).

PART B

7. Design, develop and implement program to construct Predictive / LL(1) Parsing Table for the grammar rules: $A \rightarrow aBa$, $B \rightarrow bB|Ɛ$. Use this table to parse the sentence: abba$
8. Design, develop and implement program to demonstrate Shift Reduce Parsing technique for the grammar rules: $E \rightarrow E + T|T \;\; T \rightarrow T * F|F$, F→ $(E)|id$ and parse the sentence: id + id * id.
9. Design, develop and implement syntax-directed definition of “if E then S1” and “if E then S1 else S2”
10. Write a yacc program that accepts a regular expression as input and produce its parse tree as output.
11. Design, develop and implement a program to generate the machine code using Triples for the statement A = -B * (C +D) whose intermediate code in three-address form:

T1 = -B
T2 = C + D
T3 = T1 + T2
A = T3

Course Outcomes

At the end of the course, the student should be able to

- Implement and demonstrate Lexer’s and Parser’s
- Evaluate different algorithms required for management, scheduling, allocation and communication used in operating system.

Conduct of Practical Examination:

- Experiment distribution
    - For laboratories having only one part: Students are allowed to pick one experiment from the lot with equal opportunity.
    - For laboratories having PART A and PART B: Students are allowed to pick one experiment from PART A and one experiment from PART B, with equal opportunity.
- Change of experiment is allowed only once and marks allotted for procedure to be made zero of the changed part only.
- Marks Distribution (Coursed to change in accordance with university regulations)
- For laboratories having only one part – Procedure + Execution + Viva-Voce: 15+30+5 = 50 Marks
- For laboratories having PART A and PART B

  i. Part A – Procedure + Execution + Viva = 7 + 20 + 3 = 30 Marks

  ii. Part B – Procedure + Execution + Viva = 6 + 12 + 2= 20 Marks

Operating System and UNIX Programming Lab

[As per Choice Based Credit System (CBCS) scheme]

(Effective from the academic year 2019-2020)
SEMESTER – III

Subject Code	18CSL66	**CIE Marks**	50
Number of Lecture Hours/Week	03	**SEE Marks**	50
Total Number of Lecture Hours	48	**Exam Hours**	03

CREDITS – 02

Course objectives: This course will enable students

- To learn the fundamentals of Operating Systems.
- To learn the mechanisms of OS to handle processes and threads and their communication
- To learn the mechanisms involved in memory management in contemporary OS
- To gain knowledge on distributed operating system concepts that includes architecture, Mutual exclusion algorithms, deadlock detection algorithms and agreement protocols
- To know the components and management aspects of concurrency management
- To learn programmatically to implement simple OS mechanisms

PART A

1. Implementation of CPU Scheduling Algorithms.
2. Implementation of Semaphores.
3. Implementation of Shared memory and IPC.
4. Implementation of Bankers Algorithm for Deadlock Avoidance.
5. Implementation of Deadlock Detection Algorithm.
6. Implementation of Threading and Synchronization Applications.
7. Implementation of the following Memory Allocation Methods for fixed partition.
8. Implementation of Paging Technique of Memory Management.
9. Implementation of the various File Organization Techniques.
10. Implementation of the following Page Replacement Algorithms.

PART B

1. Design a program that creates a zombie and then calls system to execute the ps command to verify that the process is zombie.

2. Design a program which demonstrates interprocess communication between a reader process and a writer process. Use mkfifo, open, read, write and close APIs in your program.
3. Design a program to illustrate the race condition.
4. Design a program that creates a zombie and then calls system to execute the ps command to verify that the process is zombie.
5. Design a program to avoid zombie process by forking twice.
6. Design a program to implement the system function.
7. Design a program to set up a real-time clock interval timer using the alarm API.

Course Outcomes

At the end of the course, the student should be able to

- Compare the performance of various CPU Scheduling Algorithms
- Implement Deadlock avoidance and Detection Algorithms
- Implement Semaphores
- Create processes and implement IPC
- Analyze the performance of the various Page Replacement Algorithms
- Implement File Organization and File Allocation Strategies

Conduct of Practical Examination:

- Experiment distribution
  - For laboratories having only one part: Students are allowed to pick one experiment from the lot with equal opportunity.
  - For laboratories having PART A and PART B: Students are allowed to pick one experiment from PART A and one experiment from PART B, with equal opportunity.
- Change of experiment is allowed only once and marks allotted for procedure to be made zero of the changed part only.
- Marks Distribution (Courseed to change in accoradance with university regulations)
- For laboratories having only one part – Procedure + Execution + Viva-Voce: 15+30+5 = 50 Marks
- For laboratories having PART A and PART B

  iii. Part A – Procedure + Execution + Viva = 7 + 20 + 3 = 30 Marks

  iv. Part B – Procedure + Execution + Viva = 6 + 12 + 2= 20 Marks

PYTHON PROGRAMMING LAB

[As per Choice Based Credit System (CBCS) scheme]

(Effective from the academic year 2019-2020)
SEMESTER – VI

Subject Code	18CSL67	**CIE Marks**	50
Number of Lecture Hours/Week	02	**SEE Marks**	50
Total Number of Lecture Hours	48	**Exam Hours**	03

CREDITS – 02

Objectives:

- To be able to introduce core programming basics and program design with functions using Python programming language.
- To understand a range of Object-Oriented Programming, as well as in-depth data and information processing techniques.
- To understand the high-performance programs designed to strengthen the practical expertise.

1. Write a program to demonstrate different number data types in Python. and perform different Arithmetic Operations on numbers in Python.
2. Write a program to create, concatenate and print a string and accessing sub-string from a given string.
3. Write a python script to print the current date in the following format “Sun May 29 02:26:23 IST 2017”
4. Write a program to create, append, and remove lists in python.
5. Write a program to demonstrate working with tuples in python.
6. Write a program to demonstrate working with dictionaries in python.
7. Write a python program to define a module and import a specific function in that module to another program.
8. Using Regular expressions, develop a Python program to
    - Identify a word with a sequence of one upper case letter followed by lower case letters.

  - Find all the patterns of “1(0+)1” in a given string.

9. Write a program that inputs a text file. The program should print all of the unique words in the file in alphabetical order.
10. Write a script named copyfile.py. This script should prompt the user for the names of two text files. The contents of the first file should be input and written to the second file.

Course Outcomes

After studying this course, students will be able to:

- Understand the basic concepts scripting and the contributions of scripting language.
- Ability to explore python especially the object-oriented concepts, and the built-in objects of Python.
- Ability to apply the knowledge to create practical and contemporary applications.

Conduct of Practical Examination:

- Experiment distribution
  - For laboratories having only one part: Students are allowed to pick one experiment from the lot with equal opportunity.
  - For laboratories having PART A and PART B: Students are allowed to pick one experiment from PART A and one experiment from PART B, with equal opportunity.
- Change of experiment is allowed only once and marks allotted for procedure to be made zero of the changed part only.
- Marks Distribution (Course to change in accordance with university regulations)
- For laboratories having only one part – Procedure + Execution + Viva-Voce: 15+30+5 = 50Marks

OPERATING SYSTEMS

[As per Choice Based Credit System (CBCS) scheme]
(Effective from the academic year 2018-2019)
SEMESTER – V

Subject Code	**18CS631/18IS631**	**CIE Marks**	**50**
Number Lecture Hour/Week	**03**	**SEE Marks**	**50**
Number of Lecture Hours	**48**	**Exam Hours**	**03**

CREDITS-02

Course objectives:

- Introduce concepts and terminology used in OS
- Explain threading and multithreaded systems
- Illustrate process synchronization and concept of Deadlock
- Introduce Memory and Virtual memory management,
- File system and storage techniques

Modules	**Teaching Hours and RBT Levels**
Module -1	
Introduction to operating systems, System structures: What operating systems do; Computer System organization; Computer System architecture; Operating System structure; Operating System operations; Process management; Memory management; Storage management; Protection and Security; Distributed system; Special-purpose systems; Computing environments. Operating System Services; User - Operating System interface; System calls; Types of system calls; System programs; Operating system design and implementation; Operating System structure; Virtual machines; Operating System generation; System boot. **Process Management** Process concept; Process scheduling; Operations on processes; Inter process communication	10 Hours L1,L2
Module -2	
Multi-threaded Programming: Overview; Multithreading models; Thread Libraries; Threading issues. Process Scheduling: Basic concepts; Scheduling Criteria; Scheduling Algorithms; Multiple-processor scheduling; Thread scheduling.	10 Hours

Process Synchronization: Synchronization: The critical section problem; Peterson's solution; Synchronization hardware; Semaphores; Classical problems of synchronization; Monitors.	L1,L2
Module -3	
Deadlocks : Deadlocks; System model; Deadlock characterization; Methods for handling deadlocks; Deadlock prevention; Deadlock avoidance; Deadlock detection and recovery from deadlock. **Memory Management:** Memory management strategies: Background; Swapping; Contiguous memory allocation; Paging; Structure of page table; Segmentation.	10 Hours L1,L2
Module-4	
Virtual Memory Management: Background; Demand paging; Copy-on-write; Page replacement; Allocation of frames; Thrashing. **File System,Implementation of File System:** File system: File concept; Access methods; Directory structure; File system mounting; File sharing; Protection: Implementing File system: File system structure; File system implementation;Directory implementation; Allocation methods; Free space management.	10 Hours L1, L2
Module-5	
Secondary Storage Structures, Protection: Mass storage structures; Disk structure; Disk attachment; Disk scheduling; Disk management; Swap space management. Protection: Goals of protection, Principles of protection, Domain of protection, Access matrix, Implementation of access matrix, Access control, Revocation of access rights, Capability- Based systems. **Case Study: The Linux Operating System:** Linux history; Design principles; Kernel modules; Process management; Scheduling; Memory Management; File systems, Input and output; Inter-process communication	08 Hours L1,L2
Course outcomes: **At the end of the course the student will be able to:** **CO1:** Demonstrate need for OS and different types of OS	

CO2: Discuss suitable techniques for management of different resources

CO3: Illustrate processor, memory, storage and file system commands

CO4: Explain the different concepts of OS in platform of usage through case studies

Text Books:

1. Abraham Silberschatz, Peter Baer Galvin, Greg Gagne, Operating System Principles 7th edition, Wiley-India, 2006.

Reference Books:

1. Ann McHoes Ida M Fylnn, Understanding Operating System, Cengage Learning, 6th Edition

2. D.M Dhamdhere, Operating Systems: A Concept Based Approach 3rd Ed, McGraw-Hill, 2013.

3. P.C.P. Bhatt, An Introduction to Operating Systems: Concepts and Practice 4th Edition, PHI(EEE), 2014.

4. William Stallings Operating Systems: Internals and Design Principles, 6th Edition, Pearson.

SOFTWARE TESTING

[As per Choice Based Credit System (CBCS) scheme]
(Effective from the academic year 2018-2019)
SEMESTER – VI

Subject Code	18IS61/18CS632	**CIE Marks**	50
Number of Lecture Hours/Week	03	**SEE Marks**	50
Total Number of Lecture Hours	50	**Exam Hours**	03

CREDITS – 04

Course objectives: This course will enable students

- Discuss test cases for any given problem
- Compare the different testing techniques
- Illustrate the problem into suitable testing model
- Understand the appropriate technique for the design of flow graph.
- Design and Develop appropriate document for the software artefact.

Module I	**Teaching Hours**
Basics of Software Testing: Basic definitions, Software Quality, Requirements, Behavior and Correctness, Correctness versus Reliability, Testing and Debugging, Test cases, Insights from a Venn diagram, Identifying test cases, Test-generation Strategies, Test Metrics, Error and fault taxonomies , Levels of testing, Testing and Verification, Static Testing. **Problem Statements:** Generalized pseudocode, the triangle problem, the NextDate function, the commission problem, the SATM (Simple Automatic Teller Machine) problem, the currency converter, Saturnwindshield wiper. **RBT: L1, L2, L3.**	10Hours
Module II	
Functional Testing: Boundary value analysis, Robustness testing, Worst-case testing, Robust Worst testing for triangle problem, Nextdate problem and commission problem,	10 Hours

Equivalence classes, Equivalence test cases for the triangle problem, NextDate function, and the commission problem, Guidelines and observations, Decision tables, Test cases for the triangle problem, NextDate function, and the commission problem, Guidelines and observations. **Fault Based Testing:** Overview, Assumptions in fault based testing, Mutation analysis, Fault-based adequacy criteria, Variations on mutation analysis. **RBT: L1, L2, L3.**	
Module III	
Structural Testing: Overview, Statement testing, Branch testing, Condition testing , Path testing: DD paths, Test coverage metrics, Basispath testing, guidelines and observations, Data –Flow testing: Definition-Use testing, Slicebasedtesting, Guidelines and observations**.** **Test Execution:** Overview of test execution, from test case specification to test cases, Scaffolding, Generic versus specific scaffolding, Test oracles, Self-checks as oracles, Capture and replay **RBT: L1, L2, L3.**	08 Hours
Module IV	
Process Framework : Basic principles: Sensitivity, redundancy, restriction, partition, visibility, Feedback, the quality process, Planning and monitoring, Quality goals, Dependability properties ,Analysis Testing, Improving the process, Organizational factors. **Planning and Monitoring the Process:** Quality and process, Test and analysis strategies and plans, Risk planning, monitoring the process, Improving the process, the quality team **Documenting Analysis and Test:** Organizing documents, Test strategy document, Analysis and test plan, Test design specifications documents, Test and analysis reports. **RBT: L1, L2, L3.**	10 Hours
Module V	
Integration and Component-Based Software Testing: Overview, Integration testing strategies, Testing components and assemblies. System, Acceptance and	10 Hours

Regression Testing: Overview, System testing, Acceptance testing, Usability, Regression testing, Regression test selection techniques, Test case prioritization and selective execution. **Levels of Testing, Integration Testing:** Traditional view of testing levels, Alternative life-cycle models, The SATM system, Separating integration and system testing, A closer look at the SATM system, Decomposition-based, call graph-based, Path-based integrations.

RBT: L1, L2, L3.

Course Outcomes

After studying this course, students will be able to:
CO 1: Discuss test cases for any given problem

CO 2: . Compare the different testing techniques

CO 3: Illustrate the problem into suitable testing model

CO 4: Understand the appropriate technique for the design of flow graph.

CO5: Design and Develop appropriate document for the software artefact

.

Question paper pattern:

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

Text Books:

1. Paul C. Jorgensen: Software Testing, A Craftsman’s Approach, 3rd Edition, Auerbach Publications, 2008. (Listed topics only from Chapters 1, 2, 5, 6, 7, 9, 10, 12, 13)
2. Mauro Pezze, Michal Young: Software Testing and Analysis – Process, Principles and Techniques, Wiley India, 2009. (Listed topics only from Chapters 3, 4, 16, 17, 20,21,22,24)
3. Aditya P Mathur: Foundations of Software Testing, Pearson Education, 2008.(Listed topics only from Section 1.2 , 1.3, 1.4 ,1.5, 1.8,1.12,6. 2.1,6. 2.4)

Reference Books:

1. Software testing Principles and Practices – Gopalaswamy Ramesh, SrinivasanDesikan,

2nd Edition, Pearson, 2007.

2. Software Testing – Ron Patton, 2nd edition, Pearson Education, 2004.

3. The Craft of Software Testing – Brian Marrick, Pearson Education, 1995.

4. AnirbanBasu, Software Quality Assurance, Testing and Metrics, PHI, 2015.

5. NareshChauhan, Software Testing, Oxford University press.

Cryptograph, network security & Cyber Law

[As per Choice Based Credit System (CBCS) scheme]
(Effective from the academic year 2019 -2020)
SEMESTER – VI

Subject Code	18CS623/IS623	**CIE Marks**	50
Number of Lecture Hours/Week	03	**SEE Marks**	50
Total Number of Lecture Hours	48	**Exam Hours**	03

CREDITS – 04

Course objectives: This course will enable students to

- To learn the concepts of cryptography
- Illustrate key management issues and solutions.
- Familiarize with cryptography and very essential algorithms.
- Introduce cyber law and ethics to be followed

Module I	**Teaching Hours**	**RBT Level**
INTRODUCTION TO NETWORK SECURITY: OSI security architecture, security attacks, security services, Security Mechanisms, a model of Network Security. **SYMMETRIC CIPHERS:** Classical Encryption Techniques, Block Ciphers and the Data Encryption Standard, Introduction to Finite Fields, Confidentiality using Symmetric Encryption.	10	L1, L2, L3
Module II		
PUBLIC - KEY ENCRYPTION AND HASH FUNCTIONS: Introduction to Number Theory, Public-Key Cryptography and RSA, Key Management : Diffie-Hellman Key Exchange, Message Authentication and Hash Functions, secure Hash Algorithm, Digital Signatures and Authentication Protocols.	10	L1, L2, L3
Module III		
NETWORK SECURITY PRACTICE: Authentication Applications: Kerberos, X.509 Authentication Service, Electronic mail Security: Pretty Good Privacy, S/MIME, IP Security: Overview, Architecture, Authentication header, ESP, Key management.	10	L1, L2, L3
Module IV		

SYSTEM SECURITY: Malicious Software: Viruses and Related Threats, Viruses Countermeasures. Distributed Denial of Service Attacks, Firewalls: Firewall Design Principles, Trusted Systems	10	L1, L2, L3
Module V		
IT act aim and objectives, Scope of the act, Major Concepts, Important provisions, Attribution, acknowledgement, and dispatch of electronic records, Secure electronic records and secure digital signatures, Regulation of certifying authorities: Appointment of Controller and Other officers, Digital Signature certificates, Duties of Subscribers, Penalties and adjudication, The cyber regulations appellate tribunal, Offences, Network service providers not to be liable in certain cases, Miscellaneous Provisions.	8	L1, L2, L3
Course Outcomes		
The students should be able to: • Discuss the cryptography and its need to various applications • Design and Develop simple cryptography algorithms • Understand the cyber security and need cyber Law		
Question paper pattern: The question paper will have ten questions. There will be 2 questions from each module. Each question will have questions covering all the topics under a module. The students will have to answer 5 full questions, selecting one full question from each module.		
Text Books: 1. William Stallings, “Cryptography and Network Security – Principles and Practices”, Pearson Education, Fourth Edition, 2006. (Chapters: 1.2, 1.3, 1.4, 1.5, 1.6, 2, 3.1, 3.2, 3.3, 4, 7, 8, 9, 10.1, 10.2, 11, 12.1, 13, 14.1, 14.2, 15, 16.1,16.2, 16.3, 16.4, 16.6, 19, 20). 2. Cryptography, Network Security and Cyber Laws – Bernard Menezes, Cengage Learning, 2010 edition (Chapters-25)		
Reference Books: 1. Atul Kahate, “Cryptography and Network Security”, Tata McGraw Hill, 2003. 2. Behrouz A. Forouzan, Introduction to Cryptography and Network Security, 2008, McGraw-Hill 3. Charles B. Pfleeger, Shari Lawrence Pfleeger, “Security in Computing”, Fourth Edition, Pearson Education, 2007. 4. Cyber Law simplified- VivekSood, Mc-GrawHill, 11th reprint , 2013 5. Cyber security and Cyber Laws, Alfred Basta, Nadine Basta, Mary brown, ravindrakumar, Cengage learning		

Rapid programming application Using Python

[As per Choice Based Credit System (CBCS) scheme]
(Effective from the academic year 2018-2019)
SEMESTER – VI

Subject Code	**18CS/IS631**	**CIE Marks**	**50**
Number Lecture Hour/Week	**03**	**SEE Marks**	**50**
Number of Lecture Hours	**48**	**Exam Hours**	**03**

CREDITS-03

Course objectives:

- Learn the syntax and semantics of Python programming language.
- Illustrate the process of structuring the data using lists, tuples and dictionaries.
- Demonstrate the use of built-in functions to navigate the file system.
- Implement the Object Oriented Programming concepts in Python.
- Appraise the need for working with various documents like Excel, PDF, Word and Others.

Modules	**Teaching Hours**
Module -1	
Python Basics: Entering Expressions into the Interactive Shell, The Integer, Floating-Point, and String Data Types, String Concatenation and Replication, Storing Values in Variables, Your First Program, Dissecting Your Program, Flow control, Boolean Values, Comparison Operators, Boolean Operators, Mixing Boolean and Comparison Operators, Elements of Flow Control, Program Execution, Flow Control Statements, Importing Modules, Ending a Program Early with sys.exit(), Functions, def Statements with Parameters, Return Values and return Statements, The None Value, Keyword Arguments and print(), Local and Global Scope, The global Statement, Exception Handling, A Short Program: Guess the Number Textbook 1: Chapters 1 – 3 RBT: L1, L2	10 Hours
Module -2	
Lists, The List Data Type, Working with Lists, Augmented Assignment Operators, Methods, Example Program: Magic 8 Ball with a List, List-like Types: Strings and Tuples,References, Dictionaries and Structuring Data, The Dictionary Data Type, Pretty Printing, Using Data Structures to Model Real-World Things, Manipulating Strings, Working with Strings, Useful String Methods, Project: Password Locker, Project: Adding Bullets to Wiki Markup Textbook 1: Chapters 4 – 6 RBT: L1, L2, L3	10 Hours
Module -3	
Pattern Matching with Regular Expressions, Finding Patterns of Text Without Regular Expressions, Finding Patterns of Text with Regular Expressions,More Pattern Matching with Regular Expressions, Greedy and Nongreedy Matching, The findall() Method, Character Classes, Making Your Own Character Classes, The Caret and Dollar Sign Characters, The Wildcard Character, Review of Regex Symbols, Case-Insensitive Matching, Substituting Strings with the sub() Method, Managing Complex Regexes, Combining re .IGNORECASE, re .DOTALL, and re .VERBOSE, Project: Phone Number and Email Address Extractor, Reading and Writing Files, Files and File Paths, The os.path Module, The File Reading/Writing Process, Saving Variables with the shelve Module,Saving Variables with the pprint.pformat() Function, Project: Generating Random Quiz Files, Project: Multiclipboard, Organizing Files, The shutil Module, Walking a Directory Tree, Compressing Files with the	8 Hours

zipfile Module, Project: Renaming Files with American-Style Dates to European-Style Dates,Project: Backing Up a Folder into a ZIP File, Debugging, Raising Exceptions, Getting the Traceback as a String, Assertions, Logging, IDLE‟s Debugger. Textbook 1: Chapters 7 – 10 RBT: L1, L2, L3	
Module-4	
Classes and objects, Programmer-defined types, Attributes, Rectangles, Instances as return values, Objects are mutable, Copying, Classes and functions, Time, Pure functions, Modifiers, Prototyping versus planning, Classes and methods, Object-oriented features, Printing objects, Another example, A more complicated example,The init method, The __str__ method, Operator overloading, Type-based dispatch, Polymorphism, Interface and implementation,Inheritance, Card objects, Class attributes, Comparing cards, Decks, Printing the deck, Add, remove, shuffle and sort, Inheritance, Class diagrams, Data encapsulation. Textbook 2: Chapters 15 – 18 RBT: L1, L2, L3	10 Hours
Module-5	
Web Scraping, Project: MAPIT.PY with the webbrowser Module, Downloading Files from the Web with the requests Module, Saving Downloaded Files to the Hard Drive, HTML, Parsing HTML with the BeautifulSoup Module, Project: “I‟m Feeling Lucky” Google Search,Project: Downloading All XKCD Comics, Controlling the Browser with the selenium Module, Working with Excel Spreadsheets, Excel Documents, Installing the openpyxl Module, Reading Excel Documents, Project: Reading Data from a Spreadsheet, Writing Excel Documents, Project: Updating a Spreadsheet, Setting the Font Style of Cells, Font Objects, Formulas, Adjusting Rows and Columns, Charts, Working with PDF and Word Documents, PDF Documents, Project: Combining Select Pages from Many PDFs, Word Documents, Working with CSV files and JSON data, The csv Module, Project: Removing the Header from CSV Files, JSON and APIs, The json Module, Project: Fetching Current Weather Data. Textbook 1: Chapters 11 – 14 RBT: L1, L2, L3	10 Hours

Course outcomes:

At the end of the course the student will be able to:

CO 1: Demonstrate proficiency in handling of loops and creation of functions.

CO 2: Identify the methods to create and manipulate lists, tuples and dictionaries.

CO 3: Discover the commonly used operations involving regular expressions and file system

CO 4: Interpret the concepts of Object-Oriented Programming as used in Python.

CO5: Determine the need for scraping websites and working with CSV, JSON and other file formats.

Text Books:

1. Al Sweigart,“Automate the Boring Stuff with Python”,1 stEdition, No Starch Press, 2015. (Available under CC-BY-NC-SA license at https://automatetheboringstuff.com/) (Chapters 1 to 18)
2. Allen B. Downey, “Think Python: How to Think Like a Computer Scientist, 2ndEdition, Green Tea Press, 2015. (http://greenteapress.com/thinkpython2/thinkpython2.pdf) (Chapters 15, 16, 17)(Download pdf files from the above links.

Reference Books:

1. Gowrishankar S, Veena A, “Introduction to Python Programming”, 1 st Edition, CRC Press/Taylor & Francis, 2018. ISBN-13: 978-0815394372
2. Jake VanderPlas, “Python Data Science Handbook: Essential Tools for Working with Data”, 1 st Edition, O‟Reilly Media, 2016. ISBN-13: 978-1491912058

3. Charles Dierbach, “Introduction to Computer Science Using Python”, 1 st Edition, Wiley India Pvt Ltd, 2015. ISBN-13: 978-8126556014
4. Wesley J Chun, “Core Python Applications Programming”, 3 rd Edition, Pearson Education India, 2015. ISBN-13: 978-9332555365

Computer Vision

[As per Choice Based Credit System (CBCS) scheme]
(Effective from the academic year 2019 -2020)
SEMESTER – VI

Subject Code	18CS644/IS644	**CIE Marks**	50
Number of Lecture Hours/Week	03	**SEE Marks**	50
Total Number of Lecture Hours	48	**Exam Hours**	03

CREDITS – 04

Course objectives: This course will enable students to

- To understand the fundamentals of image formation.
- To understand major ideas, methods and techniques of computer vision.
- To develop an appreciation for various issues in the design of computer vision and object recognition systems
- To provide programming experience from implementing computer vision and object recognition applications.

Module I	**Teaching Hours**	**RBT Level**
Image Formation Models: Monocular imaging system, Orthographic& Perspective Projection, Camera model and Camera calibration, Binocular imaging systems.	10	L1, L2, L3

Module II

Image Processing and Feature Extraction:Image representations (continuous and discrete), Edge detection	10	L1, L2, L3
Module III		
Motion Estimation:Regularization theory, Opticalcomputation, StereoVision, Motionestimation, Structure from motion	10	L1, L2, L3
Module IV		
Shape Representation and Segmentation:Deformable curves and surfaces, Snakes and active contours, Level set representations, Fourier and wavelet descriptors, Medialrepresentations, Multiresolution analysis	09	L1, L2, L3
Module V		
Object recognition:Hough transforms and other simple object recognition methods, Shape correspondence and shape matching, Principal Component analysis, Shape priors for recognition.	09	L1, L2, L3
Course Outcomes		
The students should be able to: • identify basic concepts, terminology, theories, models and methods in the field of computer vision, · • Describe known principles of human visual system, · • Describe basic methods of computer vision related to multi-scale representation, edge detection and detection of other primitives, stereo, motion and object recognition, · • suggest a design of a computer vision system for a specific problem		
Question paper pattern: The question paper will have ten questions. There will be 2 questions from each module. Each question will have questions covering all the topics under a module. The students will have to answer 5 full questions, selecting one full question from each module.		
Text Books: Computer Vision - A modern approach, by D. Forsyth and J. Ponce, Prentice Hall Robot Vision, by B. K. P. Horn, McGraw-Hill.		
Reference Books: 1. Richard Szeliksy "Computer Vision: Algorithms and Applications" (http://szeliski.org/Book/) 2. Haralick& Shapiro, "Computer and Robot Vision", Vol II 3. G_erardMedioni and Sing Bing Kang "Emerging topics in computer vision"		

4. Emanuele Trucco and AllessandroVerri "Introductory Techniques for 3-D Computer Vision", Prentice Hall, 1998.
5. Olivier Faugeras, "Three-Dimensional Computer Vision", The MIT Press, 1993.

PROBABILITY STATISTICS AND QUEUING THEORY

[As per Choice Based Credit System (CBCS) scheme]
(Effective from the academic year 2018-2019)
SEMESTER – VI

Subject Code	18CS/IS634	**CIE Marks**	50
Number of Lecture Hours/Week	03	**SEE Marks**	50
Total Number of Lecture Hours	48	**Exam Hours**	03

CREDITS – 03

Course objectives: This course will enable students

- Develop analytical capability and impart knowledge of Statistics and queuing probability.
- Apply above concepts in Engineering and Technology.
- Acquire knowledge of Hypothesis testing and Queuing methods and their applications so as to enable them to apply them for solving real world problems

MODULE I	**Teaching Hours**
Axioms of probability, Conditional probability, Total probability, Baye's theorem, Discrete Random variable, Probability mass function, Continuous Random variable. Probability density function, Cumulative Distribution	10Hours

Function, and its properties, Two-dimensional Random variables, Joint pdf / cdf and their properties **RBT: L1, L2, L3.**	
Module II	
Probability Distributions / Discrete distributions: Binomial, Poisson Geometric and hyper-geometric distribution and their properties.continous distribution.uniform, Normal, exponential distributions and their properties **RBT: L1, L2, L3.**	10 Hours
Module III	
Random Processes: Classification, methods of description , special classes, Average value of random processes, analytical representation of random processes, Autocorrelation function, cross-correlation function and their properties, Ergodicity, Poisson process, Markov Process, Markov chain **RBT: L1, L2, L3.**	08 Hours
Module IV	
Testing Hypothesis : Testing of Hypothesis: Formulation of Null Hypothesis, Critical region, level of significance, errors in testing, Tests of significance for Large and Small Samples, t-distribution, its properties and uses, F-distribution, its properties and uses, Chi-square distribution, its properties and uses, $\chi 2$ – test for goodness of fit, $\chi 2$ test for Independence	10 Hours
Module V	
Symbolic Representation of a Queuing Model, Poisson Queue system, Little Law, Types of Stochastic Processes, Birth-Death Process, The M/M/1 Queuing System, The M/M/s Queuing System, The M/M/s Queuing with Finite buffers. **RBT: L1, L2, L3.**	10 Hours
Course Outcomes	
After studying this course, students will be able to: CO 1: Demonstrate use of probability and characterize probability models using probability mass (density) functions & cumulative distribution functions. CO 2: Explain the techniques of developing discrete & continuous probability distributions and its applications. CO 3: Describe a random process in terms of its mean and correlation functions. CO 4: Outline methods of Hypothesis testing for goodness of fit.	

CO 5: Define the terminology &nomenclature appropriate queuing theory and also distinguish various queuing models.

Question paper pattern:

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

Text Books:

1. Probability, Statistics and Queuing Theory, V. Sundarapandian, Eastern Economy Edition,PHI Learning Pvt. Ltd, 2009

1. **Reference Books:** Probability & Statistics with Reliability, Queuing and Computer Applications, 2nd Edition by Kishor. S. Trivedi , Prentice Hall of India ,2004.

2. Probability, Statistics and Random Processes, 1st Edition by P Kausalya, Pearson Education, 2013.

SOFTWARE ENGINEERING

[As per Choice Based Credit System (CBCS) scheme]

(Effective from the academic year 2019 -2020)

SEMESTER – VI

Subject Code	18CS/IS641	**CIE Marks**	50
Number of Lecture Hours/Week	03	**SEE Marks**	50
Total Number of Lecture Hours	48	**Exam Hours**	03

Module 1	**Teaching Hours**
Introduction: Need for Software Engineering, Professional Software Development, Software Engineering Ethics. Case Studies. **Software Processes**: Models: Waterfall Model, Incremental Model and Spiral Model, Process activities. **Requirements Engineering**: Requirements Engineering Processes**,** Functional and non-functional requirements**,** The software Requirements Document, Requirements Specification, Requirements validation, Requirements Management	**11 Hours**
Module 2	
System Models: Context models **,** Interaction models, Structural models, Behavioral models, Model-driven engineering . **Architectural Design**: Architectural design decisions, Architectural patterns. **Design and Implementation**: Object-oriented design using the UML **,** Design patterns, Implementation issues **.**	**11 Hours**

Module 3	
Software Testing: Development testing , Test-driven development , Release testing, User testing. **Software Evolution**: Evolution processes, Program evolution dynamics, Software maintenance, Legacy system management.	**9 Hours**
Module 4	
Project Planning: Software pricing, Plan-driven development. Project scheduling : Estimation techniques .**Quality management**: Software quality, Reviews and inspections Software measurement and metrics, Software standards **.**	**09 Hours**
Module 5	
Managing People: Selecting staff; Motivating people; Managing people; The People Capability Maturity Model. Software Cost Estimation: Productivity; Estimation techniques; Algorithmic cost modeling, Project duration and staffing. **Agile Software Development:** The Agile Manifesto: Values and Principles. Agile methods: SCRUM (Ref "The SCRUM Primer, Ver 2.0") and Extreme Programming ,Plan-driven and agile development.	**10 Hours**

Course Outcomes: After studying this course, students will be able to:

- Apply the software engineering lifecycle.
- Analyze and specify software requirements.
- Design a software system, component, or process to meet desired needs within realistic constraints.
- Assess professional and ethical responsibility
- Function on multi-disciplinary teams
- Make use of techniques, skills, and modern engineering tools necessary for engineering practice
- Comprehend software systems or parts of software systems.

Text Books:

1. Ian Sommerville: Software Engineering, 9th Edition, Pearson Education, 2012.
2. The SCRUM Primer, Ver 2.0, http://www.goodagile.com/scrumprimer/scrumprimer20.pdf

Reference Books:

1. Roger S. Pressman: Software Engineering-A Practitioners approach, 7th Edition, Tata McGraw Hill**.**
2. Pankaj Jalote: An Integrated Approach to Software Engineering, Wiley India

Web Reference for eBooks on Agile:

1. http://agilemanifesto.org/
2. http://www.jamesshore.com/Agile-Book/

Multicore Architecture

[As per Choice Based Credit System (CBCS) scheme]
(Effective from the academic year 2019 -2020)
SEMESTER – VI

Subject Code	18CS642	**CIE Marks**	50
Number of Lecture Hours/Week	03	**SEE Marks**	50
Total Number of Lecture Hours	48	**Exam Hours**	03

CREDITS – 04

Course objectives: This course will enable students to

To Become familiar with the concepts of computer networks

- To understand technologies of multicore architecture and performance measures
- Demonstrate problems related to multiprocessing
- Illustrate windows threading, posix threads, openmp programming
- Analyze the common problems in parallel programming

Module I	**Teaching Hours**	**RBT Level**
Introduction to Multi-core Architecture: Motivation for Concurrency in software, Parallel Computing Platforms, Parallel Computing in Microprocessors, Differentiating Multi-core Architectures from Hyper-Threading Technology, Multi-threading on Single-Core versus Multi-Core	10	L1, L2, L3

Platforms Understanding Performance, Amdahl’s Law, Growing Returns: Gustafson’s Law. **System Overview of Threading :** Defining Threads, System View of Threads, Threading above the Operating System, Threads inside the OS, Threads inside the Hardware, What Happens When a Thread Is Created, Application Programming Models and Threading, Virtual Environment: VMs and Platforms, Runtime Virtualization, System Virtualization.		
Module II		
Fundamental Concepts of Parallel Programming: Designing for Threads, Task Decomposition, Data Decomposition, Data Flow Decomposition, Implications of Different Decompositions, Challenges You’ll Face, Parallel Programming Patterns, A Motivating Problem: Error Diffusion, Analysis of the Error Diffusion **Algorithm, An Alternate Approach:** Parallel Error Diffusion, Other Alternatives.Threading and Parallel Programming Constructs: Synchronization, Critical Sections, Deadlock, Synchronization Primitives, Semaphores, Locks, Condition Variables, Messages, Flow Control- based Concepts, Fence, Barrier, Implementation-dependent Threading Features	10	L1, L2, L3
Module III		
Threading APIs : ThreadingAPIs for Microsoft Windows, Win32/MFC Thread APIs, Threading APIs for Microsoft. NET Framework, Creating Threads, Managing Threads, Thread Pools, Thread Synchronization, POSIX Threads, Creating Threads, Managing Threads, Thread Synchronization, Signaling, Compilation and Linking.	9	L1, L2, L3
Module IV		
OpenMP: A Portable Solution for Threading : Challenges in Threading a Loop, Loop-carried Dependence, Data-race Conditions, Managing Shared and Private Data, Loop Scheduling and Portioning, Effective Use of Reductions, Minimizing Threading Overhead, Work-sharing Sections, Performance-oriented Programming, Using Barrier and No wait, Interleaving Single-thread and Multithread Execution, Data Copy-in and Copy-out, Protecting Updates of Shared Variables, Intel Task queuing Extension to OpenMP, OpenMP Library	9	L1, L2, L3

Functions, OpenMP Environment Variables, Compilation, Debugging, performance		
Module V		
Solutions to Common Parallel Programming Problems : Too Many Threads, Data Races, Deadlocks, and Live Locks, Deadlock, Heavily Contended Locks, Priority Inversion, Solutions for Heavily Contended Locks, Non-blocking Algorithms, ABA Problem, Cache Line Ping-ponging, Memory Reclamation Problem, Recommendations, Thread-safe Functions and Libraries, Memory Issues, Bandwidth, Working in the Cache, Memory Contention, Cache-related Issues, False Sharing, Memory Consistency, Current IA-32 Architecture, Itanium Architecture, High-level Languages, Avoiding Pipeline Stalls on IA-32,Data Organization for High Performance.	10	L1, L2, L3
Course Outcomes		
The students should be able to: • Identify the issues involved in multicore architectures • Explain fundamental concepts of parallel programming and its design issues • Solve the issues related to multiprocessing and suggest solutions • Discuss salient features of different multicore architectures and how they exploit parallelism • Illustrate OpenMP and programming concept		
Question paper pattern: The question paper will have ten questions. There will be 2 questions from each module. Each question will have questions covering all the topics under a module. The students will have to answer 5 full questions, selecting one full question from each module.		
Text Book: Multicore Programming , Increased Performance through Software Multi-threading by ShameemAkhter and Jason Roberts , Intel Press , 2006		

Network Programming

[As per Choice Based Credit System (CBCS) scheme]
(Effective from the academic year 2018-2019)
SEMESTER – VI

Subject Code	18CS/IS643	**CIE Marks**	50
Number of Lecture Hours/Week	03	**SEE Marks**	50
Total Number of Lecture Hours	50	**Exam Hours**	03

CREDITS – 04

Course objectives: This course will enable students

- Define Network Programming.
- Demonstrate programming with TCP and SCTP.
- Explain key management and routing sockets.
- Evaluate advanced Socket Programming APIs.

Module I	**Teaching Hours**
Introduction to network application: client/server communication, OSI Model, BSD Networking history, Test Networks and Hosts, Unix Standards, 64-bit architectures, Transport Layer: TCP, UDP and SCTP. **RBT: L1, L2, L3.**	10Hours

Module II	
Sockets Introduction – socket address structures, value-result arguments, byte ordering and manipulation functions, address conversion functions, Elementary TCP Sockets – socket,connect, bind, listen, accept , fork and concurrent server design, getcsockname and getpeername functions and TCP Client/Server Example- client/server programming through TCP sockets, Normal startup, termination, POSIX signal handling, Signal handling in server, Crashing, rebooting of server host, shutdown. **RBT: L1, L2, L3**	10 Hours
Module III	
I/O Multiplexing and Socket Options, Elementary SCTP Sockets- Interface Models, STCP_X functions, shutdown function, Notifications, SCTP Client/Server Examples – One to-Many, Head–of-Line Blocking, Controlling number of streams and Termination, IPv4 and IPv6 Interoperability–different interoperability scenarios. **RBT: L1, L2, L3.**	08 Hours
Module IV	
Daemon Processes, syslogd, daemonizing functions and the inetd super server, Advanced I/O functions- readv, writev, sendmsg and recvmsg, Ancillary data, Advanced polling, Unix domain protocols- socket address structure, functions and communication scenarios, Nonblocking I/O – connect and accept examples. **RBT: L1, L2, L3.**	10 Hours
Module V	
IOCTL operations- socket, file, interface configuration information, ARP cache and routing table operations, Routing sockets- data link socket address structure, reading and writing, sysctl operations, interface name and index functions, Key Management functions –reading, writing, SADB, SA, Dynamically Maintaining SA's, Out-of-Band data, Threads-basic thread functions, TCP echo server using threads, Mutexes and Conditional variables.	10 Hours
Course Outcomes	
After studying this course, students will be able to: • Develop applications that communicate with each other using TCP and SCTP. • Identify the IPv4 and IPv6 compatibility. • Evaluate socket programming APIs.	
Question paper pattern: • The question paper will have ten questions. • There will be 2 questions from each module. • Each question will have questions covering all the topics under a module.	

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

Text Books:

- W. Richard Stevens, Bill Fenner, Andrew M. Rudoff: “UNIX Network Programming”. Volume 1, Third Edition, Pearson 2004.

Reference Books:

- Barry Nance: “Network Programming in C”, PHI 2002 3.Bob Quinn, Dave Shute: “Windows Socket Network Programming”, Pearson 2003.
- Richard Stevens: “UNIX Network Programming”. Volume 2, Second Edition.

MOBILE COMPUTING

[As per Choice Based Credit System (CBCS) scheme]
(Effective from the academic year 2018-2019)
SEMESTER – VI

Subject Code	**18CS/IS644**	**CIE Marks**	**50**
Number Lecture Hour/Week	**03**	**SEE Marks**	**50**
Number of Lecture Hours	**48**	**Exam Hours**	**03**

CREDITS-02

Course objectives:

This course will enable students to:

- Define concepts of wireless communication.
- Compare and contrast propagation methods, Channel models, capacity calculations multiple antennas and multiple user techniques used in the mobile communication.
- Explain CDMA, GSM. Mobile IP, WImax and Different Mobile OS
- Illustrate various Markup Languages CDC, CLDC, MIDP; Programming for CLDC, MIDlet model and security concerns

Modules	**Teaching Hours**

<table>
<tr><td colspan="2"><b>Module -1</b></td></tr>
<tr><td>Mobile Computing Architecture: Architecture for Mobile Computing, 3-tier Architecture, Design Considerations for Mobile Computing. Emerging Technologies: Wireless broadband (WiMAX), Mobile IP: Introduction, discovery, Registration, Tunneling, Cellular IP, Mobile IP with IPv6. Wireless Networks : Global Systems for Mobile Communication (GSM): GSM Architecture, Entities, Call routing in GSM, PLMN Interface, GSM Addresses and Identities, Network Aspects in GSM, Mobility Management, GSM Frequency allocation. Short Service Messages (SMS): Introduction to SMS, SMS Architecture, SMMT, SMMO, SMS as Information bearer, applications
<b>Textbook1: 2.4 - 2.6, 4.4 - 4.6, 5, 6.</b>
<b>RBT: L1, L2</b></td><td>10 Hours</td></tr>
<tr><td colspan="2"><b>Module -2</b></td></tr>
<tr><td>GPRS and Packet Data Network, GPRS Network Architecture, GPRS Network Operations, Data Services in GPRS, Applications for GPRS, Billing and Charging in GPRS. Spread Spectrum technology, IS-95, CDMA versus GSM, Wireless Data, Third Generation Networks, Applications on 3G, Mobile Client: Moving beyond desktop, Mobile handset overview, Mobile phones and their features, PDA, Design Constraints in applications for handheld devices.
<b>Textbook 1: 7,9.2 - 9.7, 12.2 - 12.6</b>
<b>RBT: L1, L2</b></td><td>10 Hours</td></tr>
<tr><td colspan="2"><b>Module -3</b></td></tr>
<tr><td>Mobile OS and Computing Environment: Smart Client Architecture, The Client: User Interface, Data Storage, Performance, Data Synchronization, Messaging. The Server: Data Synchronization, Enterprise Data Source, Messaging. Mobile Operating Systems: WinCE, Palm OS, Symbian OS, Linux, Proprietary OS Client Development: The development process, Need analysis phase, Design phase, Implementation and Testing phase, Deployment phase, Development Tools, Device Emulators
<b>Textbook 2: 7, 8.</b>
<b>RBT: L1, L2</b></td><td>10 Hours</td></tr>
<tr><td colspan="2"><b>Module-4</b></td></tr>
<tr><td>Building Wireless Internet Applications: Thin client overview: Architecture, the client, Middleware, messaging Servers, Processing a Wireless request, Wireless Applications Protocol (WAP) Overview, Wireless Languages: Markup Languages, HDML, WML, 10
Hours HTML, cHTML, XHTML, VoiceXML.
<b>Textbook 2: 11, 12, 13</b>
<b>RBT: L1, L2</b></td><td>10 Hours</td></tr>
<tr><td colspan="2"><b>Module-5</b></td></tr>
<tr><td>J2ME: Introduction, CDC, CLDC, MIDP; Programming for CLDC, MIDlet model, Provisioning, MIDlet life-cycle, Creating new application, MIDlet event handling, GUI in MIDP, Low level GUI Components, Multimedia APIs; Communication in MIDP, Security Considerations in MIDP.
<b>Textbook 1: 15.1 - 15.10</b>
<b>RBT: L1, L2</b></td><td>08 Hours</td></tr>
<tr><td colspan="2"><b>Course outcomes:</b>
At the end of the course the student will be able to:
• Explain state of art techniques in wireless communication.
• Discover CDMA, GSM. Mobile IP, WImax</td></tr>
</table>

- Demonstrate program for CLDC, MIDP let model and security concerns

Textbooks:

1. Ashok Talukder, Roopa Yavagal, Hasan Ahmed: Mobile Computing, Technology, Applications and Service Creation, 2nd Edition, Tata McGraw Hill, 2010.
2. Martyn Mallik: Mobile and Wireless Design Essentials, Wiley India, 2003

Reference Books:

1. Raj kamal: Mobile Computing, Oxford University Press, 2007.
2. Iti Saha Misra: Wireless Communications and Networks, 3G and Beyond, Tata McGraw Hill, 2009.

ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING
[As per Choice Based Credit System (CBCS) scheme]
(Effective from the academic year 2018-2019)
SEMESTER – VII

Subject Code	**18CS71**	**CIE Marks**	**50**
Number of Lecture Hours/Week	**03**	**SEE Marks**	**50**
Total Number of Lecture Hours	**48**	**Exam Hours**	**03**

CREDITS – 04

Course objectives: This course will enable students

- To Understand the Basic principles of Artificial Intelligence.
- Become familiar with AI toward problem solving, inference, perception, knowledge representation, and learning.
- To interpret the different supervised classification methods and tree-based models
- To understand concept learning, ANN, Bayes classifier, k nearest neighbor.

Module I	**Teaching Hours**
What is artificial intelligence? Problems, problem spaces and search, Heuristic search techniques **Textbook 1: Chapter 1, 2 and 3** **RBT: L1, L2**	10Hours
Module II	
Knowledge representation issues, Predicate logic, Representation knowledge using rules. Concept Learning: Concept learning task, Concept learning as search, Find-S algorithm, Candidate Elimination Algorithm, Inductive bias of Candidate Elimination Algorithm. **Texbook 1: Chapter 4, 5 and 6** **Texbook2: Chapter 2 (2.1-2.5, 2.7)** **RBT: L1, L2, L3**	10 Hours
Module III	
Decision Tree Learning: Introduction, Decision tree representation, Appropriate problems, ID3 algorithm. Artificial Neural Network: Introduction, NN representation, Appropriate problems, Perceptions, Back propagation algorithm. **Texbook2: Chapter 3 (3.1-3.4), Chapter 4 (4.1-4.5)** **RBT: L1, L2, L3**	08 Hours
Module IV	
Bayesian Learning: Introduction, Bayes theorem, Bayes theorem and concept learning, ML and LS error hypothesis, ML for predicting, MDL principle, Bates optimal classifier, Gibbs algorithm, Navie Bayes classifier, BBN, EM Algorithm **Texbook2: Chapter 6** **RBT: L1, L2, L3**	10 Hours
Module V	

Instance-Base Learning: Introduction, k-Nearest Neighbor Learning, Locally weighted regression, Radial basis function, Case-Based reasoning. Reinforcement Learning: Introduction, The learning task, Q-Learning. **Texbook 1: Chapter 8 (8.1-8.5), Chapter 13 (13.1 – 13.3)** **RBT: L1, L2, L3**	10 Hours

Course Outcomes

After studying this course, students will be able to:

- CO 1: Demonstrate fundamental understanding of artificial intelligence (AI) and expert systems.
- CO 2: Apply basic principles of AI in solutions that require problem solving, inference, perception, knowledge representation, and learning.
- CO 3: Apply the different supervised learning methods and tree based models.
- CO 4: Illustrate concept learning, ANN, Bayes classifier, k nearest neighbor.

Question paper pattern:

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

Text Books:

1. Elaine Rich, Kevin K and S. B. Nair, “Artificial Inteligence”, 3rd Edition, McGraw Hill Education, 2017.
2. Tom M Mitchell,“Machine Lerning”,1st Edition, McGraw Hill Education, 2017.

Reference Books:

1. Saroj Kaushik, Artificial Intelligence, Cengage learning
2. Stuart Rusell, Peter Norving , Artificial Intelligence: A Modern Approach, Pearson Education 2nd Edition
3. AurÈlienGÈron,"Hands-On Machine Learning with Scikit-Learn and Tensor Flow: Concepts, Tools, and Techniques to Build Intelligent Systems", 1st Edition, Shroff/O'Reilly Media, 2017.
4. Trevor Hastie, Robert Tibshirani, Jerome Friedman,The Elements of Statistical Learning, 2nd edition, springer series in statistics.
5. EthemAlpaydın, Introduction to machine learning, second edition, MIT press.
6. Srinvivasa K G and Shreedhar, ― Artificial Intelligence and Machine Learning‖, Cengage

BIG DATA AND ANALYTICS
[As per Choice Based Credit System(CBCS) scheme]
(Effective from the academic year 2018 -2019)
SEMESTER – VII

Course Code	**18CS72**	**CIE Marks**	50
Number of Contact Hours/Week	03	**SEE Marks**	50
Total Number of Contact Hours	48	**Exam Hours**	03

CREDITS –4

Course Learning Objectives: This course will enable students to:

- Understand Hadoop Distributed File system and examine MapReduce Programming
- Explore Hadoop tools and manage Hadoop with Ambari
- Appraise the role of Business intelligence and its applications across industries
- Assess core data mining techniques for data analytics
- Identify various Text Mining techniques

Module – I	**Teaching Hours**
Hadoop Distributed File System Basics, Running Example Programs and Benchmarks, Hadoop MapReduce Framework, MapReduce Programming **Textbook 1:Chapter 3,4,5,6** **RBT:L1,L2**	**10 Hours**
Module – II	**10 Hours**
Essential Hadoop Tools, Business Intelligence Concepts and Application, Data Warehousing, Data Mining **Textbook 1:Chapter 7** **Textbook 2:Chapter 2,3,4** **RBT:L1,L2,L3**	
Module – III	**10 Hours**
Data Visualization, Decision Trees, Regression, Artificial Neural Networks **Textbook 2:Chapter 5,6,7,8** **RBT:L1,L2,L3**	
Module – IV	**10 Hours**
Cluster Analysis, Association Rule Mining, Text Mining, Naïve-Bayes Analysis **Textbook 2:Chapter 9,10,11,12** **RBT:L1,L2,L3**	
Module – V	**10 Hours**
Support Vector Machines, Web Mining, Social Network Analysis **Textbook 2:Chapter 13,14,15** **RBT:L1,L2,L3**	

Course outcomes: The students should be able to:

- Master the concepts of HDFS and MapReduce framework
- Investigate Hadoop related tools for Big Data Analytics and perform basic Hadoop Administration
- Recognize the role of Business Intelligence, Data warehousing and Visualization in decision making
- Infer the importance of core data mining techniques for data analytics
- Compare and contrast different Text Mining Techniques

Question paper pattern:

- The queston paper will have ten questions.
- There will be 2 questions from each module.
- Each question will have questions covering all the topics under a module.
- The students will have to answer5 full questions,Selecting one full question from each module.

Text Books:

1. Douglas Eadline,"Hadoop 2 Quick-Start Guide: Learn the Essentials of Big Data Computing in the Apache Hadoop 2 Ecosystem", 1st Edition, Pearson Education, 2016. ISBN-13: 978-9332570351
2. Anil Maheshwari, “Data Analytics”, 1st Edition, McGraw Hill Education,2017. ISBN-13: 978-9352604180

Reference Books:

1) Tom White, “Hadoop: The Definitive Guide”, 4 Edition, O’Reilly Media,
2) Boris Lublinsky, Kevin T.Smith, Alexey Yakubovich,"Professional Hadoop Solutions", 1st Edition, Wrox Press, 2014ISBN-13: 978-8126551071
3) Eric Sammer,"Hadoop Operations: A Guide for Developers and Administrators",1st Edition, O'Reilly Media, 2012.ISBN-13: 978-9350239261

ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING LABORATORY

[As per Choice Based Credit System (CBCS) scheme]

(Effective from the academic year 2018 -2019)

SEMESTER – VII

Subject Code	18CSL75	**CIE Marks**	50
Number of Lecture Hours/Week	02+06	**SEE Marks**	50
Total Number of Lecture Hours	40	**Exam Hours**	03

CREDITS – 02

Course objectives: This course will enable students

- Implement and evaluate AI and ML algorithms in and Python programming language.
- Understand the evaluation of different algorithms.

1. Implement A* Search algorithm.

2. Implement AO* Search algorithm.

3. For a given set of training data examples stored in a .CSV file, implement and demonstrate the Candidate-Elimination algorithm to output a description of the set of all hypotheses consistent with the training examples.

4 Write a program to demonstrate the working of the decision tree based ID3 algorithm. Use an appropriate data set for building the decision tree and apply this knowledge toclassify a new sample.

5. Build an Artificial Neural Network by implementing the Backpropagation algorithm and test the same using appropriate data sets.

6. Write a program to implement the naïve Bayesian classifier for a sample training data set stored as a .CSV file. Compute the accuracy of the classifier, considering few test data sets.

7.Apply EM algorithm to cluster a set of data stored in a .CSV file. Use the same data set for clustering using k-Means algorithm. Compare the results of these two algorithms and comment on the quality of clustering. You can add Java/Python ML library classes/API in the program.

8. Write a program to implement k-Nearest Neighbour algorithm to classify the iris data set. Print both correct and wrong predictions. Java/Python ML library classes can be used for this problem.
9. Implement the non-parametric Locally Weighted Regressional gorithm in order to fit data points. Select appropriate data set for your experiment and draw graphs

Course Outcomes

The students should be able to:

- Implement and demonstrate AI and ML algorithms.
- Evaluate different algorithms.

BIG DATA AND ANALYTICS LABORATORY
(Effective from the academic year 2020 -2021)
SEMESTER – VII

Subject Code	18CSL76	**CIE Marks**	50
Number of Lecture Hours/Week	02+06	**SEE Marks**	50
Total Number of Lecture Hours	40	**Exam Hours**	03

CREDITS-02

Course Objectives: This course will enable students to

Implement and evaluate BIG DATA AND ANALYTICS Problems

1. Installation of Single Node Hadoop Cluster on Ubuntu
2. Hadoop Programming: Word Count MapReduce Program Using Eclipse
3. Implementing Matrix Multiplication Using One Map-Reduce Step.
4. Implementing Relational Algorithm on Pig.
5. Implementing database operations on Hive.
6. Implementing Bloom Filter using Map-Reduce
7. Implementing Frequent Item set algorithm using Map-Reduce.
8. Implementing Clustering algorithm using Map-Reduce
9. Implementing Page Rank algorithm using Map-Reduce

Course Outcomes

The Students should be able to:

- Implement and demonstrate BIG DATA AND ANALYTICS Problems.
- Evaluate different problems.

AWS FUNDAMENTALS LABORATORY

[As per Choice Based Credit System (CBCS) scheme]
(Effective from the academic year 2018 -2019)
SEMESTER – VII

Subject Code	18CSL77	**CIE Marks**	50
Number of Lecture Hours/Week	02+06	**SEE Marks**	50
Total Number of Lecture Hours	40	**Exam Hours**	03

CREDITS – 02

Course objectives: This course will enable students
Designed to give you a comprehensive understanding of the foundational services offered by AWS including compute, storage, networking, database, and identity & access management (IAM).

PART A

Illustration of the following services.

- Application Auto Scaling
- Amazon Aurora
- AWS Cloud9
- Amazon CloudFormation
- Amazon CloudFront
- AWS CloudShell
- AWS CloudTrail
- Amazon CloudWatch
- AWS CodeCommit
- Amazon Cognito
- Amazon Comprehend
- AWS DeepRacer
- Amazon DynamoDB
- Amazon EC2 Auto Scaling
- AWS Elastic Beanstalk
- Amazon Elastic Block Store (EBS)
- Amazon Elastic Compute Cloud (EC2)
- Amazon Elastic Container Registry (ECR)
- Amazon Elastic File System (EFS)
- Amazon Elastic Inference
- Elastic Load Balancing
- Amazon EventBridge
- Amazon Forecast
- AWS Glue
- AWS Glue DataBrew
- AWS Identity and Access Management (IAM)
- AWS Key Management Service (KMS)
- AWS Lambda
- Amazon Lex
- Amazon Lightsail
- Amazon Marketplace Subscriptions (Amazon ML)
- Amazon Polly
- Amazon Rekognition
- Amazon Relational Database Service (RDS)
- AWS Resource Groups & Tag Editor
- AWS RoboMaker
- Amazon SageMaker
- AWS Secrets Manager
- AWS Security Token Service (STS)
- AWS Service Catalog
- Amazon Simple Notification Service (SNS)
- Amazon Simple Queue Service (SQS)
- Amazon Simple Storage Service (S3)
- Amazon Simple Storage Service Glacier (S3 Glacier)
- AWS Step Functions
- AWS Systems Manager (SSM)
- Amazon Textract
- Amazon Translate
- AWS Trusted Advisor
- Amazon Virtual Private Cloud (Amazon VPC)
- AWS Well-Architected Tool

PART B

1. Introduction to AWS IAM
2. Build Your VPC and Launch a Web Server
3. Introduction to Amazon EC2
4. Working with Amazon EBS
5. Build Your DB Server and Interact with Your DB Using an App.

Course Outcomes

The students should be able to:

- Increased Enterprise Cloud Migration to AWS
- AWS Is the Fastest Growing Public Cloud Service
- AWS Skills Demand Is Outstripping Supply

INDUSTRIAL PSYCHOLOGY AND ORGANISATIONAL BEHAVIOUR

B.Tech, VII Semester, Electronics & Communication Engineering

[As per Choice Based Credit System (CBCS) scheme]

Subject Code	18HSM79	CIE Marks	50
Number of Lecture Hour/Week	01	SEE Marks	50
Total Number of Lecture Hours	20	Exam Hours	03

CREDITS-01

Course Objectives: This course will enable students to:

1. Relating human psychology to science
2. Understand the human psychology
3. Understand the nature of organization and organization models
4. Understand the human social communication
5. Understand the leadership qualities

Modules	**Teaching Hours**
Module -1	
Introduction to I/O psychology: Major fields of I/O psychology, brief history of I/O psychology, employment of I/O psychology, ethics in I/O psychology. (Chapter-1) **RBT L1,L2**	**3 Hours**
Module -2	
Organisational communication: Types of organizational communication, interpersonal communication, improving employee communication skills. (Chapter-11) **RBT L1,L2**	**3 Hours**
Module -3	
Leadership : Introduction, personal characteristics associated with leadership, interaction between the leadership and the situation specific leader skills, leadership where we are today. (Chapter-12) **RBT L1,L2**	**5 Hours**
Module -4	
Group behaviour- teams and conflicts Group dynamics, factors affecting group performance, individual versus group performance, group conflicts. (Chapter-13) **RBT L1,L2**	**5 Hours**
Module-5	
Stress management: Dealing with the demands of life and work, stress defined, predisposition to stress, sources of stress, consequences of stress, stress reduction intervention related to life /work issues. (Chapter-15) **RBT L1,L2**	**4 Hours**

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

1. Comprehend the knowledge and concepts of human psychology
2. Know the importance of psychology
3. Have insight into individual and group behavior
4. Deal with people in better way
5. Motivate groups and build groups

Text Book: Michael G.Aamodt, Industrial/Organizational Psychology: An Applied Approach, 6th Edition, Wadsworth Cengage Learning, ISBN: 978-0-495-60106-7.

Reference Books:

1. Blum M.L. Naylor J.C., Horper & Row, Industrial Psychology, CBS Publisher, 1968
2. Luthans, Organizational Behaviour, McGraw Hill, International, 1997
3. Morgan C.t.,King R.A.,John Rweisz &John Schoples, Introduction to Psychology, McHraw Hill, 1966
4. Schermerhorn J.R.Jr., Hunt J.G &Osborn R.N., Managing, Organizational Behaviour, John Willy

DATA MINING AND DATA WAREHOUSING

[As per Choice Based Credit System (CBCS) Scheme]

SEMESTER-VII

Subject Code	18CS731	**CIE Marks**	50
Number Lecture Hour/Week	3L	**SEE Marks**	50
Number of Lecture Hours	40	**Exam Hours**	03

CREDITS-03

Course Objectives

- Define multi-dimensional data models.
- Explain rules related to association, classification and clustering analysis.
- Compare and contrast between different classification and clustering algorithms

Module -1	Teaching Hours
Data Warehousing & modeling: Basic Concepts: Data Warehousing: A multitier Architecture, Data warehouse models: Enterprise warehouse, Data mart and virtual warehouse, Extraction, Transformation and loading, Data Cube: A multidimensional data model, Stars, Snowflakes and Fact constellations: Schemas for multidimensional Data models, Dimensions: The role of concept Hierarchies, Measures: Their Categorization and computation, Typical OLAP Operations Textbook 2: Ch.4.1,4.2 RBT: L1, L2, L3	08 Hours
Module -2	
Data warehouse implementation& Data mining: Efficient Data Cube computation: An overview, Indexing OLAP Data: Bitmap index and join index, Efficient processing of OLAP Queries, OLAP server Architecture ROLAP versus MOLAP Versus HOLAP. : Introduction: What is data mining, Challenges, Data Mining Tasks, Data: Types of Data, Data Quality, Data Preprocessing, Measures of Similarity and Dissimilarity. Textbook 2: Ch.4.4 Textbook 1: Ch.1.1,1.2,1.4, 2.1 to 2.4 RBT: L1, L2, L3	08 Hours
Module -3	
Association Analysis: Association Analysis: Problem Definition, Frequent Item set Generation, Rule generation. Alternative Methods for Generating Frequent Item sets, FPGrowth Algorithm, Evaluation of Association Patterns. Textbook 1: Ch 6.1 to 6.7 (Excluding 6.4) RBT: L1, L2, L3	08 Hours
Module -4	
Classification: Decision Trees Induction, Method for Comparing Classifiers, Rule Based Classifiers, Nearest Neighbor Classifiers, Bayesian Classifiers. Textbook 1: Ch 4.3,4.6,5.1,5.2,5.3 RBT: L1, L2, L3	08 Hours
Module -5	
Clustering Analysis: Overview, K-Means, Agglomerative Hierarchical Clustering, DBSCAN, Cluster Evaluation, Density-Based Clustering, Graph-Based Clustering, Scalable Clustering Algorithms. Textbook 1: Ch 8.1 to 8.5, 9.3 to 9.5 RBT: L1, L2, L3	08 Hours

Course Outcomes: After studying this course, students will be able to:

- Identify data mining problems and implement the data warehouse
- Write association rules for a given data pattern.
- Choose between classification and clustering solution.

Text Books:

1. Pang-Ning Tan, Michael Steinbach, Vipin Kumar: Introduction to Data Mining, Pearson, First impression,2014.
2. Jiawei Han, Micheline Kamber, Jian Pei: Data Mining -Concepts and Techniques, 3rd Edition, Morgan Kaufmann Publisher, 2012.

Reference Books:

1. Sam Anahory, Dennis Murray: Data Warehousing in the Real World, Pearson,Tenth Impression,2012.
2. Michael J Berry, Gordon S Linoff: Mastering Data Mining , Wiley Edition, second edtion,2012.

DATA MINING AND DATA WAREHOUSING

[As per Choice Based Credit System (CBCS) Scheme]

SEMESTER-VII

Subject Code	18CS732	**CIE Marks**	50
Number Lecture Hour/Week	3L	**SEE Marks**	50
Number of Lecture Hours	40	**Exam Hours**	03

CREDITS-03

Course Objectives

- Define multi-dimensional data models.
- Explain rules related to association, classification and clustering analysis.
- Compare and contrast between different classification and clustering algorithms

Module -1	Teaching Hours
Data Warehousing & modeling: Basic Concepts: Data Warehousing: A multitier Architecture, Data warehouse models: Enterprise warehouse, Data mart and virtual warehouse, Extraction, Transformation and loading, Data Cube: A multidimensional data model, Stars, Snowflakes and Fact constellations: Schemas for multidimensional Data models, Dimensions: The role of concept Hierarchies, Measures: Their Categorization and computation, Typical OLAP Operations Textbook 2: Ch.4.1,4.2 RBT: L1, L2, L3	08 Hours
Module -2	
Data warehouse implementation& Data mining: Efficient Data Cube computation: An overview, Indexing OLAP Data: Bitmap index and join index, Efficient processing of OLAP Queries, OLAP server Architecture ROLAP versus MOLAP Versus HOLAP. : Introduction: What is data mining, Challenges, Data Mining Tasks, Data: Types of Data, Data Quality, Data Preprocessing, Measures of Similarity and Dissimilarity. Textbook 2: Ch.4.4 Textbook 1: Ch.1.1,1.2,1.4, 2.1 to 2.4 RBT: L1, L2, L3	08 Hours
Module -3	
Association Analysis: Association Analysis: Problem Definition, Frequent Item set Generation, Rule generation. Alternative Methods for Generating Frequent Item sets, FPGrowth Algorithm, Evaluation of Association Patterns. Textbook 1: Ch 6.1 to 6.7 (Excluding 6.4) RBT: L1, L2, L3	08 Hours
Module -4	
Classification: Decision Trees Induction, Method for Comparing Classifiers, Rule Based Classifiers, Nearest Neighbor Classifiers, Bayesian Classifiers. Textbook 1: Ch 4.3,4.6,5.1,5.2,5.3 RBT: L1, L2, L3	08 Hours
Module -5	
Clustering Analysis: Overview, K-Means, Agglomerative Hierarchical Clustering, DBSCAN, Cluster Evaluation, Density-Based Clustering, Graph-Based Clustering, Scalable Clustering Algorithms. Textbook 1: Ch 8.1 to 8.5, 9.3 to 9.5 RBT: L1, L2, L3	08 Hours

Course Outcomes: After studying this course, students will be able to:

- Identify data mining problems and implement the data warehouse
- Write association rules for a given data pattern.
- Choose between classification and clustering solution.

Text Books:

1. Pang-Ning Tan, Michael Steinbach, Vipin Kumar: Introduction to Data Mining, Pearson, First impression,2014.
2. Jiawei Han, Micheline Kamber, Jian Pei: Data Mining -Concepts and Techniques, 3rd Edition, Morgan Kaufmann Publisher, 2012.

Reference Books:

1. Sam Anahory, Dennis Murray: Data Warehousing in the Real World, Pearson,Tenth Impression,2012.
2. Michael J Berry, Gordon S Linoff: Mastering Data Mining , Wiley Edition, second edtion,2012.

AWS CLOUD FOUNDATIONS

[As per Choice Based Credit System(CBCS) scheme]
(Effective from the academic year 2018 -2019)
SEMESTER – VII

Course Code	**18CS732**	**CIE Marks**	50
Number of Contact Hours/Week	03	**SEE Marks**	50
Total Number of Contact Hours	40	**Exam Hours**	03

CREDITS –4

Course Learning Objectives: This course will enable students to:

- Describe the security and compliance measures of the AWS Cloud, including AWS Identity and Access Management (IAM)
- Create a virtual private cloud (VPC) by using Amazon Virtual Private Cloud (Amazon VPC)
- Demonstrate when to use Amazon Elastic Compute Cloud (Amazon EC2), AWS Lambda, and AWS Elastic Beanstalk
- Differentiate between Amazon Simple Storage Service (Amazon S3), Amazon Elastic Block Store (Amazon EBS), Amazon Elastic File System (Amazon EFS), and Amazon Simple Storage Service Glacier (Amazon S3 Glacier)
- Demonstrate when to use AWS database services, including Amazon Relational Database Service (Amazon RDS), Amazon DynamoDB, Amazon Redshift, and Amazon Aurora
- Explain the architectural principles of the AWS Cloud
- Explore key concepts related to Elastic Load Balancing, Amazon CloudWatch, and Amazon EC2Auto Scaling

Module – I	**Teaching Hours**
Cloud Concepts Overview: Introduction to Cloud Computing, Advantages of the Cloud, Introduction to AWS, Moving to the AWS Cloud, **Cloud Economics and Billing**: Introduction Fundamentals of Pricing, Total Cost of Ownership, Simple Monthly Calculator, Delaware North Case Study, AWS Organizations, AWS Billing and Cost Management, Billing Dashboard	**08 Hours**
Module – II	
AWS Global Infrastructure Overview: Introduction, AWS Global Infrastructure, AWS Global Infrastructure, AWS Services and Service Categories, AWS Management Console Clickthrough. **Cloud Security:** Introduction, AWS Shared Responsibility Model, AWS Shared Responsibility Model, AWS IAM, AWS IAM Console Demonstration, Securing a New AWS Account, Introduction to AWS IAM, Securing Accounts, Securing Data, Working to Ensure Compliance	**08 Hours**
Module – III	
Networking and Content Delivery: Introduction, Networking Basics, Amazon VPC, VPC Networking, Label This Diagram, Amazon VPC Console Demonstration, VPC Security, Design a VPC Build a VPC and Launch a Web Server, Route 53, CloudFront. **Compute:** Introduction, Compute Services Overview, Amazon EC2 Part 1, Amazon EC2 Part 2, Amazon EC2 Part 3, Introduction to Amazon EC2, Amazon EC2 versus Managed Services, Amazon	**08 Hours**

EC2 Part Console Demonstration, Amazon EC2 Cost Optimization, Container Services, Introduction to AWS Lambda, AWS Lambda, Introduction to AWS Elastic Beanstalk, AWS Elastic Beanstalk	
Module – IV	
Storage: Introduction, AWS EBS, Amazon Elastic Block Store Console, Demonstration, Working with EBS, AWS S3, AWS S3 Console Demonstration, AWS EFS, AWS EFS Console Demonstration, AWS S3 Glacier, AWS S3 Glacier Console Demonstration, Storage Technology Selection**. Databases:** Introduction, Amazon RDS, Amazon RDS Console Demonstration, Build a Database Server, Amazon DynamoDB, Amazon DynamoDB Demonstration, Amazon Redshift, Amazon Aurora, Database Case Study	**08 Hours**
Module – V	
Cloud Architecture: Introduction, AWS Well-Architected Framework Design, Principles, AWS Well-Architected Framework Design, Principles, Operational Excellence, Security, Reliability Performance Efficiency, Cost Optimization, Reliability & High Availability, AWS Trusted Advisor, Interpret AWS Trusted Advisor Recommendations. **Automatic Scaling and Monitoring:** Introduction, Elastic Load Balancing, Elastic Load Balancing, Amazon CloudWatch, Amazon CloudWatch, Amazon EC2 Auto Scaling, Scale & Load Balance your Architecture	**08 Hours**

Course outcomes: The students should be able to:

- Learned the fundamentals of building secure, performance efficient, reliable, operationally excellent, and cost-optimized services in the cloud.
- Learned about cost optimization techniques like right sizing, serverless, reservations, and spot instances.
- Learned about reviewing, tracking, and optimizing your budget using services like the Cost Explorer, tags, and budgets.

Web links and Video Lectures:

https://awsacademy.instructure.com/courses/3515/modules

Reference Books:

1) Mark Wilkins, “Learning Amazon Web Services (AWS): A Hands-On Guide to the Fundamentals of AWS Cloud”, Publisher(s): Addison-Wesley Professional, O’Reilly Media 2019.
2) “Mastering AWS Cost Optimization: Real-world technical and operational cost-saving best practices (Second Edition)”, by Eli Mansoor and Yair Green 2020

System Modelling and Simulation

[As per Choice Based Credit System(CBCS) scheme]
(Effective from the academic year 2018 -2019)
SEMESTER – VII

Course Code	**18CS733**	**CIE Marks**	50
Number of Contact Hours/Week	03	**SEE Marks**	50
Total Number of Contact Hours	40	**Exam Hours**	03

CREDITS –4

Course Learning Objectives: This course will enable students to:

- Explain the basic system concept and definitions of system;
- Discuss techniques to model and to simulate various systems;
- Analyze a system and to make use of the information to improve the performance

Module – I	**Teaching Hours**
Introduction: When simulation is the appropriate tool and when it is not appropriate, Advantages and disadvantages of Simulation; Areas of application, Systems and system environment; Components of a system; Discrete and continuous systems, Model of a system; Types of Models, Discrete-Event System Simulation Simulation examples: Simulation of queuing systems. General Principles. **Textbook 1: Ch. 1, 2, 3.1.1, 3.1.3 RBT: L1, L2, L3**	**08 Hours**
Module – II	
Statistical Models in Simulation: Review of terminology and concepts, Useful statistical models, Discrete distributions. Continuous distributions, Poisson process, Empirical distributions. **Queuing Models**: Characteristics of queuing systems, Queuing notation, Long-run measures of performance of queuing systems, Long-run measures of performance of queuing systems, Steady-state behavior of M/G/1 queue, Networks of queues **Textbook 1: Ch. 5,6.1 to 6.3, 6.4.1,6.6 RBT: L1, L2, L3**	**08 Hours**
Module – III	
Random-NumberGeneration: Properties of random numbers; Generation of pseudo-random numbers, Techniques for generating random numbers,Tests for Random Numbers, Random **Variate Generation:** Inverse transform technique Acceptance-Rejection technique. **Textbook 1: Ch. 7,8.1, 8.2 RBT: L1, L2, L3**	**08 Hours**
Module – IV	
Input Modeling: Data Collection; Identifying the distribution with data, Parameter estimation, Goodness of Fit Tests, Fitting a non-stationary Poisson process, Selecting input models without data, Multivariate and Time-Series input models. **Estimation of Absolute Performance:** Types of simulations with respect to	**08 Hours**

output analysis ,Stochastic nature of output data, Measures of performance and their estimation. **Textbook 1: Ch. 9, 11.1 to 11.3 RBT: L1, L2, L3**	
Module – V	
Measures of performance and their estimation, Output analysis for terminating simulations Output analysis for steady-state simulations. **Verification, Calibration and Validation:** Optimization: Model building, verification and validation, Verification of simulation models, Verification of simulation models, Calibration and validation of models, Optimization via Simulation. **Textbook 1: Ch. 11.4, 11.5, 10 RBT: L1, L2, L3**	**08 Hours**
Course outcomes: The students should be able to:	
• Explain the system concept and apply functional modeling method to model the activities of a static system • Describe the behavior of a dynamic system and create an analogous model for a dynamic system; • Simulate the operation of a dynamic system and make improvement according to the simulation results.	
Text Books: 1. Jerry Banks, John S. Carson II, Barry L. Nelson, David M. Nicol: Discrete-Event System Simulation, 5 th Edition, Pearson Education, 2010.	
Reference Books: 1) Lawrence M. Leemis, Stephen K. Park: Discrete – Event Simulation: A First Course, Pearson Education, 2006. 2) Averill M. Law: Simulation Modeling and Analysis, 4 th Edition, Tata McGraw-Hill, 2007	

STORAGE AREA NETWORKS

[As per Choice Based Credit System (CBCS) Scheme]

SEMESTER-VII

Subject Code	18CS734	**CIE Marks**	50
Number Lecture Hour/Week	3L	**SEE Marks**	50
Number of Lecture Hours	40	**Exam Hours**	03

CREDITS-03

Course Objectives

- Evaluate storage architectures.
- Define backup, recovery, disaster recovery, business continuity, and replication
- Examine emerging technologies including IP-SAN
- Understand logical and physical components of a storage infrastructure
- Identify components of managing and monitoring the data center
- Define information security and identify different storage virtualization technologies

Module -1	**Teaching Hours**
Storage System: Introduction to Information Storage: Information Storage, Evolution of Storage Architecture, Data Center Infrastructure, Virtualization and Cloud Computing. Data Center Environment: Application Database Management System (DBMS), Host (Compute), Connectivity, Storage, Disk Drive Components Disk Drive Performance, Host Access to Data, Direct-Attached Storage, Storage Design Based on Application Textbook1 : Ch.1.1 to 1.4, Ch.2.1 to 2.10 RBT: L1, L2	08 Hours
Module -2	
Data Protection - RAID : RAID Implementation Methods, RAID Array Components, RAID Techniques, RAID Levels, RAID Impact on Disk Performance, RAID Comparison. Intelligent Storage Systems : Components of an Intelligent Storage System, Types of Intelligent Storage Systems. Fibre Channel Storage Area Networks - Fibre Channel: Overview, The SAN and Its Evolution, Components of FC SAN. Textbook1 : Ch.3.1 to 3.6, Ch. 4.1, 4.3, Ch. 5.1 to 5.3 RBT: L1, L2	08 Hours
Module -3	
IP SAN and FCoE: iSCSI, FCIP, Network-Attached Storage: General-Purpose Servers versus NAS Devices, Benefits of NAS, File Systems and Network File Sharing, Components of NAS, NAS I/O Operation, NAS Implementations, NAS File-Sharing Protocols, Factors Affecting NAS Performance Textbook1 : Ch.6.1, 6.2, Ch. 7.1 to 7.8 RBT: L1, L2	08 Hours
Module -4	
Introduction to Business Continuity: Information Availability, BC Terminology, BC Planning Life Cycle, Failure Analysis, Business Impact Analysis, BC Technology Solutions, Backup and Archive: Backup Purpose, Backup Considerations, Backup Granularity, Recovery Considerations, Backup Methods, Backup Architecture, Backup and Restore Operations, Backup Topologies, Backup in NAS Environments Textbook1 Ch.9.1 to 9.6, Ch. 10.1 to 10.9 RBT: L1, L2	08 Hours
Module -5	

Local Replication: Replication Terminology, Uses of Local Replicas, Replica Consistency, Local Replication Technologies, Tracking Changes to Source and Replica, Restore and Restart Considerations, Creating Multiple Replicas. Remote Replication: Modes of Remote 08 Replication, Remote Replication Technologies. Securing the Storage Infrastructure: Information Security Framework, Risk Triad, Storage Security Domains. Security Implementations in Storage Networking. Textbook1: Ch.11.1 to 11.7, Ch. 12.1, 12.2, Ch. 14.1 to 14.4 RBT: L1, L2	08 Hours
Course Outcomes: After studying this course, students will be able to: • Identify key challenges in managing information and analyze different storage networking technologies and virtualization. • Explain components and the implementation of NAS. • Describe CAS architecture and types of archives and forms of virtualization. • Illustrate the storage infrastructure and management activities.	
Text Books: 3. EMC Education Services, “Information Storage and Management”, Wiley India Publications	
Reference Books: 1. Paul Massiglia, Richard Barker, "Storage Area Network Essentials: A Complete Guide to Understanding and Implementating SANs Paperback", 1st Edition, Wiley India Publications, 2008.	

INTERNET OF THINGS

[As per Choice Based Credit System (CBCS) scheme]
(Effective from the academic year 2021 -2021)
SEMESTER – VII

Subject Code	18CS741	**CIE Marks**	50
Number of Lecture Hours/Week	3L	**SEE Marks**	50
Total Number of Lecture Hours	40	**Exam Hours**	03

CREDITS –

Course objectives: This course will enable students to

- Define and explain basic issues, policy and challenges in the IoT
- Illustrate Mechanism and Key Technologies in IoT
- Explain the Standard of the IoT
- Explain resources in the IoT and deploy of resources into business.

Module I	**Teaching Hours**	**RBT Levels**
What is IoT, Genesis of IoT, IoT and Digitization, IoT Impact, Convergence of IT and IoT, IoT Challenges, IoT Network Architecture and Design, Drivers Behind New Network Architectures, Comparing IoT Architectures, A Simplified IoT Architecture, The Core IoT Functional Stack, IoT Data Management and Compute Stack.	08 Hours	L1, L2, L3
Module II		
Smart Objects: The “Things” in IoT, Sensors, Actuators, and Smart Objects, Sensor Networks, Connecting Smart Objects, Communications Criteria, IoT Access Technologies.	08 Hours	L1, L2, L3
Module III		
IP as the IoT Network Layer, The Business Case for IP, The need for Optimization, Optimizing IP for IoT, Profiles and Compliances, Application Protocols for IoT, The Transport Layer, IoT Application Transport Methods.	08 Hours	L1, L2, L3
Module IV		
Data and Analytics for IoT, An Introduction to Data Analytics for IoT, Machine Learning, Big Data Analytics Tools and Technology, Edge Streaming Analytics, Network Analytics, Securing IoT, A Brief History of OT Security, Common Challenges in OT Security, How IT and OT Security Practices and Systems Vary, Formal Risk Analysis Structures: OCTAVE and FAIR, The Phased Application of Security in an Operational Environment	08 Hours	L1, L2, L3
Module V		
IoT Physical Devices and Endpoints - Arduino UNO: Introduction to Arduino, Arduino UNO, Installing the Software, Fundamentals of Arduino Programming. IoT Physical Devices and Endpoints - RaspberryPi: Introduction to RaspberryPi, About the RaspberryPi Board: Hardware Layout, Operating Systems on RaspberryPi, Configuring RaspberryPi, Programming RaspberryPi with Python, Wireless Temperature Monitoring System Using Pi, DS18B20 Temperature Sensor, Connecting Raspberry Pi via SSH, Accessing Temperature from DS18B20 sensors, Remote access to RaspberryPi, Smart and Connected Cities, An IoT Strategy for Smarter Cities, Smart City IoT Architecture, Smart City Security Architecture, Smart City Use-Case Examples.	08 Hours	L1, L2, L3

Course Outcomes

The students should be able to:

- Interpret the impact and challenges posed by IoT networks leading to new architectural models.
- Compare and contrast the deployment of smart objects and the technologies to connect them to network
- Appraise the role of IoT protocols for efficient network communication
- Elaborate the need for Data Analytics and Security in IoT
- Illustrate different sensor technologies for sensing real world entities and identify the applications of IoT in Industry

Question paper pattern:

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

Textbooks:

1. David Hanes, Gonzalo Salgueiro, Patrick Grossetete, Robert Barton, Jerome Henry,"IoT Fundamentals: Networking Technologies, Protocols, and Use Cases for the Internet of Things", 1stEdition, Pearson Education (Cisco Press Indian Reprint). (ISBN: 9789386873743)

2. Srinivasa K G, "Internet of Things",CENGAGE Leaning India, 2017

Reference Books:

1. Vijay Madisetti and ArshdeepBahga, "Internet of Things (A Hands-on-Approach)", 1stEdition, VPT, 2014. (ISBN: 978-8173719547)

2. Raj Kamal, "Internet of Things: Architecture and Design Principles", 1st Edition, McGraw Hill Education, 2017. (ISBN: 978-9352605224)

Additional Study material & e-Books

1. NPTEL notes and Videos

BLOCKCHAIN TECHNOLOGY

[As per Choice Based Credit System (CBCS) scheme]

(Effective from the academic year 2018-2019)

SEMESTER – VII

Subject Code	**18CS742**	**CIE Marks**	**50**
Number Lecture Hour/Week	**3L**	**SEE Marks**	**50**
Number of Lecture Hours	**40**	**Exam Hours**	**03**

CREDITS-04

Course objectives:

- Define and Explain the fundamentals of Blockchain
- Illustrate the technologies of blockchain
- Decribe the models of blockchain
- Analyze and demonstrate the Ethereum

Modules	**Teaching Hours**
Module -1	
Blockchain 101: Distributed systems, History of blockchain, Introduction to blockchain, Types of blockchain, CAP theorem and blockchain, Benefits and limitations of blockchain. **Text Book 1: Chapter 1** **RBT: L1, L2**	08 Hours
Module -2	
Decentralization and Cryptography: Decentralization using blockchain, Methods of decentralization, Routes to decentralization, Decentralized organizations. Cryptography and Technical Foundations: Cryptographic primitives, Asymmetric cryptography, Public and private keys **Text Book 1: Chapter 2, Chapter 4** **RBT: L1, L2**	08 Hours
Module -3	
Bitcoin and Alternative Coins A: Bitcoin, Transactions, Blockchain, Bitcoin payments B: Alternative Coins Theoretical foundations, Bitcoin limitations, Namecoin, Litecoin, Primecoin, Zcash **Text Book 1: Chapter 3, Chapter 6, Chapter 8** **RBT: L1, L2**	08 Hours
Module-4	
Smart Contracts and Ethereum 101: Smart Contracts: Definition, Ricardian contracts.Ethereum 101: Introduction, Ethereum blockchain, Elements of the Ethereum blockchain, Precompiled contracts. **Text Book 1: Chapter 10** **RBT: L1, L2**	08 Hours
Module-5	
Alternative Blockchains: Blockchains Blockchain-Outside of Currencies: Internet of Things, Government, Health, Finance, Media **Text Book 1: Chapter 17** **RBT L1, L2**	08 Hours

Course outcomes:

At the end of the course the student will be able to:

- CO 1: Define and Explain the fundamentals of Blockchain

- CO2: Illustrate the technologies of blockchain
- CO3: Describe the models of blockchain
- CO3: Analyse and demonstrate the Ethereum.
- CO4: Analyse and demonstrate Hyperledger fabric

Text Books:

1. Mastering Blockchain - Distributed ledgers, decentralization and smart contracts explained, Imran Bashir, Packt Publishing Ltd, Second Edition, ISBN 978-1-78712-544-5, 2017.

ference Books:

1. Bitcoin and Cryptocurrency Technologies, Arvind Narayanan, Joseph Bonneau, Edward Felten, 2016.
2. Blockchain Basics: A Non-Technical Introduction in 25 Steps, Daniel Drescher, Apress, First Edition, 2017.
3. Mastering Bitcoin: Unlocking Digital Cryptocurrencies, Andreas M. Antonopoulos, O'Reilly Media, First Edition, 2014

PYTHON APPLICATION PROGRAMMING

[As per Choice Based Credit System(CBCS) scheme]
(Effective from the academic year 2018 -2019)
SEMESTER – VII
(NOT FOR CS/IS STUDENTS)

Course Code	18CS743	**CIE Marks**	50
Number of Contact Hours/Week	03	**SEE Marks**	50
Total Number of Contact Hours	40	**Exam Hours**	03

CREDITS –4

Course Learning Objectives: This course will enable students to:

- Learn Syntax and Semantics and create Functions in Python.
- Handle Strings and Files inPython.
- Understand Lists, Dictionaries and Regular expressions in Python.
- Implement Object Oriented Programming concepts in Python
- Build Web Services, Network and Database Programs in Python.

Module – I	**Teaching Hours**
Why should you learn to write programs, Variables, expressions and statements, Conditional execution, Functions **LI, L2, L3**	**08 Hours**
Module – II	
Iteration, Strings, Files, **LI, L2, L3**	**08 Hours**
Module – III	
Lists, Dictionaries, Tuples, Regular Expressions, **LI, L2, L3**	**08 Hours**
Module – IV	
Classes and objects, Classes and functions, Classes and methods, **LI, L2, L3**	**08 Hours**
Module – V	
Networked programs, Using Web Services, Using databases and SQL, **LI, L2, L3**	**08 Hours**

Course outcomes: The students should be able to:

- Examine Python syntax and semantics and be fluent in the use of Python flow control and functions.
- Demonstrate proficiency in handling Strings and File Systems.
- Create, run and manipulate Python Programs using core data structures like Lists, Dictionaries and use Regular Expressions.
- Interpret the concepts of Object-Oriented Programming as used in Python.
- Implement exemplary applications related to Network Programming, Web Services and Databases in Python.

Text Books:

1. Charles R. Severance, “Python for Everybody: Exploring Data Using Python 3”, 1 Edition, CreateSpace Independent Publishing Platform, 2016 (Chapters 1 — 13, 15).

2. Allen B. Downey “Think Python: How to Think Like a Computer 2nd Edition, Green Tea Press, 2015 (Chapters 15,16,17)

Reference Books:

4) Mark Lutz, “Programming Python”, 4t h Edition, O’Reilly Media, 2011.ISBN-13:978-9350232873.
5) Wesley J Chun, “Core Python Applications Programming”, 3rd Edition, Pearson Education India, 2015. ISBN-13: 978-9332555365.
6) Reema Thareja, “Python Programming using problem solving approach”, Oxford university press, 2017

NEURAL NETWORKS AND DEEP LEARNING

[As per Choice Based Credit System (CBCS) Scheme]

SEMESTER-VII

Subject Code	18CS744	**CIE Marks**	50
Number Lecture Hour/Week	3L	**SEE Marks**	50
Number of Lecture Hours	40	**Exam Hours**	03

CREDITS-03

Course Objectives

- Identify the deep learning algorithms which are more appropriate for various types of learning tasks in various domains.
- Implement deep learning algorithms and solve real-world problems.
- Execute performance metrics of Deep Learning Techniques

Module -1	**Teaching Hours**
Introduction to ANN: Biological to Artificial neuron, Training an MLP, Training a DNN with Tensor Flow, Fine tuning NN Hyper Parameters Up and Running with TensorFlow. **Chapter 9 and 10**	08 Hours
Module -2	
Deep Neural network: Introduction, Vanishing Gradient problems, Reusing Pretrained layers, Faster optimizers, avoiding over fitting through regularization. **Chapter 11.**	08 Hours
Module -3	
Distributing Tensor flow across devices and servers: Multiple devices on a single machine, multiple servers, parallelizing NN on a Tensor Flow cluster Convolution Neural Network: Architecture of the visual cortex, Convolutional layer, Pooling layer, CNN architecture **Chapter 12 and 13**	08 Hours
Module -4	
Recurrent Neural Network: Recurrent neurons, Basic RNN in Tensor Flow, Training RNN, Deep RNNs, LSTM Cell, GRU Cell, NLP **Chapter 14**	08 Hours
Module -5	
Autoencoders: Efficient data representation, Performing PCA, Stacked autoencoders, Unsupervised pretraining using SA, Denoising, Sparse autoencoders, variational and other autoencoders. Reinforcement Learning: Learning to optimize rewards, policy search, Introduction to Open AI Gym, Neural network polices, Evaluating actions, Policy gradients, Markov decision processes, TDL and Q-learning, Learning to play Ms.Pac-man using Deep Q Learning**. Chapter 15 and 16**	08 Hours

Course Outcomes: After studying this course, students will be able to:

- Identify the deep learning algorithms which are more appropriate for various types of learning tasks in various domains.
- Implement deep learning algorithms and solve real-world problems.
- Execute performance metrics of Deep Learning Techniques.

Text Books:

1. Hands on Machine Learning with Scikit-Learn &TensorFlow, Aurelien Geron, O'Reilly, 2019

Reference Books:

1. Deep Learning Lan Good fellow and YoshuaBengio and Aaron CourvilleMIT Press2016.

2. Neural Networks and Deep Learning, Charu C. Aggarwal, Springer International Publishing, 2018