

VISVESVARAYA TECHNOLOGICAL UNIVERSITY, BELAGAVI

B.E. in Computer Science and Engineering

Scheme of Teaching and Examinations 2021

Outcome Based Education(OBE) and Choice Based Credit System (CBCS) (Effective from the academic year 2021 - 22)

III SEMESTER

Sl. No	Course and Course Code	Course Title	Teaching Department (TD) and Question Paper Setting Board (PSB)	Teaching Hours /Week				Examination			Credits	
				Theory Lecture	Tutorial	Practical/ Drawing	Self -Study	Duration in hours	CIE Marks	SEE Marks		Total Marks
				L	T	P	S					
1	BSC 21MAT31	Transform Calculus, Fourier Series and Numerical Techniques	Maths	3	0	0		03	50	50	100	3
2	IPCC 21CS32	Data Structures and Applications	Any CS Board Department	3	0	2		03	50	50	100	4
3	IPCC 21CS33	Analog and Digital Electronics		3	0	2		03	50	50	100	4
4	PCC 21CS34	Computer Organization and Architecture		3	0	0		03	50	50	100	3
5	PCC 21CSL35	Object Oriented Programming with JAVA Laboratory		0	0	2		03	50	50	100	1
6	UHV 21UH36	Social Connect and Responsibility	Any Department	0	0	2		01	50	50	100	1
7	HSMC 21KSK37/47	Sanskritika Kannada	TD and PSB: HSMC	1	0	0		01	50	50	100	1
	HSMC 21KKBK37/47	Balake Kannada										
	OR											
	HSMC 21CIP37/47	Constitution of India and Professional Ethics										
8	AEC 21CS38X/21 CSL38X	Ability Enhancement Course - III	TD: Concerned department PSB: Concerned Board	If offered as Theory Course				01	50	50	100	1
				1	0	0						
				If offered as lab. course				02				
				0	0	2						
Total									400	400	800	18

9	Scheduled activities for III to VIII semesters	NMDC 21NS83	National Service Scheme (NSS)	NSS	All students have to register for any one of the course namely National Service Scheme, Physical Education (PE)(Sports and Athletics) and Yoga with the concerned coordinator of the course during the first week of III semester. The activities shall be carried out from (for 5 semesters) between III semester to VIII semester. SEE in the above courses shall be conducted during VIII semester examinations and the accumulated CIE marks shall be added to the SEE marks. Successful completion of the registered course is mandatory for the award of the degree. The events shall be appropriately scheduled by the colleges and the same shall be reflected in the colander prepared for the NSS, PE and Yoga activities.
		NMDC 21PE83	Physical Education (PE)(Sports and Athletics)	PE	
		NMDC 21YO83	Yoga	Yoga	

Course prescribed to lateral entry Diploma holders admitted to III semester B.E./B.Tech programs

1	NMDC 21MATDIP31	Additional Mathematics - I	Maths	02	02	--	--	---	100	---	100	0
---	-----------------	----------------------------	-------	----	----	----	----	-----	-----	-----	-----	---

Note:BSC: Basic Science Course, IPCC: Integrated Professional Core Course, PCC: Professional Core Course, INT – Internship, HSMC: Humanity and Social Science & Management Courses, AEC–Ability Enhancement Courses. UHV: Universal Human Value Course.

L –Lecture, T – Tutorial, P- Practical/ Drawing, S – Self Study Component, CIE: Continuous Internal Evaluation, SEE: Semester End Examination. TD- Teaching Department, PSB: Paper Setting department

21KSK37/47Samskrutika Kannada is for students who speak, read and write Kannada and **21KKB37/47**Balake Kannada is for non-Kannada speaking, reading, and writing students.

Integrated Professional Core Course (IPCC): Refers to Professional Theory Core Course Integrated with Practical's of the same course. Credit for IPCC can be 04 and its Teaching–Learning hours (L : T : P) can be considered as (3 : 0 : 2) or (2 : 2 : 2). The theory part of the IPCC shall be evaluated both by CIE and SEE. The practical part shall be evaluated by only CIE (no SEE). However, questions from the practical part of IPCC shall be included in the SEE question paper. For more details, the regulation governing the Degree of Bachelor of Engineering /Technology (BE/B.Tech.) 2021-22 may be referred.

21INT49 Inter/Intra Institutional Internship: All the students admitted to engineering programs under the lateral entry category shall have to undergo a mandatory 21INT49 Inter/Intra Institutional Internship of 03 weeks during the intervening period of III and IV semesters. The internship shall be slated for CIE only and will not have SEE. The letter grade earned through CIE shall be included in the IV semester grade card. The internship shall be considered as a head of passing and shall be considered for vertical progression and for the award of degree. Those, who do not take up / complete the internship shall be declared fail and shall have to complete during subsequently after satisfying the internship requirements. The faculty coordinator or mentor shall monitor the students' internship progress and interact with them for the successful completion of the internship.

Non-credit mandatory courses (NCCM):

(A) Additional Mathematics I and II:

(1) These courses are prescribed for III and IV semesters respectively to lateral entry Diploma holders admitted to III semester of B.E./B.Tech., programs. They shall attend the classes during the respective semesters to complete all the formalities of the course and appear for the Continuous Internal Evaluation (CIE). In case, any student fails to register for the said course/fails to secure the minimum 40 % of the prescribed CIE marks, he/she shall be deemed to have secured an F grade. In such a case, the student has to fulfill the course requirements during subsequent semester/s to earn the qualifying CIE marks. These courses are slated for CIE only and has no SEE.

(2) Additional Mathematics I and II shall not be considered for vertical progression as well as for the calculation of SGPA and CGPA, but completion of the courses shall be mandatory for the award of degree.

(3) Successful completion of the courses Additional Mathematics I and II shall be indicated as satisfactory in the grade card. Non-completion of the courses Additional Mathematics I and II shall be indicated as Unsatisfactory. **(B) National Service Scheme/Physical Education (Sport and Athletics)/ Yoga:**

(1) Securing 40 % or more in CIE, 35 % or more marks in SEE and 40 % or more in the sum total of CIE + SEE leads to successful completion of the registered course.

(2) In case, students fail to secure 35 % marks in SEE, they have to appear for SEE during the subsequent examinations conducted by the University.

(3) In case, any student fails to register for NSS, PE or Yoga/fails to secure the minimum 40 % of the prescribed CIE marks, he/she shall be deemed to have not completed the requirements of the course. In such a case, the student has to fulfill the course requirements during subsequent semester/s to earn the qualifying CIE marks.

(4) Successful completion of the course shall be indicated as satisfactory in the grade card. Non-completion of the course shall be indicated as Unsatisfactory.

(5) These courses shall not be considered for vertical progression as well as for the calculation of SGPA and CGPA, but completion of the courses shall be mandatory for the award of degree.

Ability Enhancement Course - III

21CSL381	Mastering Office	21CS383	
21CS382	Programming IN c++	21CS384	

VISVESVARAYA TECHNOLOGICAL UNIVERSITY, BELAGAVI

B.E. in Computer Science and Engineering

Scheme of Teaching and Examinations 2021

Outcome-Based Education (OBE) and Choice Based Credit System (CBCS) (Effective from the academic year 2021 - 22)

IV SEMESTER

Sl. No	Course and Course Code	Course Title	Teaching Department (TD) and Question Paper Setting Board (PSB)	Teaching Hours /Week				Examination			Credits	
				Theory Lecture	Tutorial	Practical/ Drawing	Self-Study	Duration in hours	CIE Marks	SEE Marks		Total Marks
				L	T	P	S					
1	BSC 21CS41	Mathematical Foundations for Computing	Maths	2	2	0		03	50	50	100	3
2	IPCC 21CS42	Design and Analysis of Algorithms	Any CS Board Department	3	0	2		03	50	50	100	4

3	IPCC 21CS43	Microcontroller and Embedded Systems		3	0	2		03	50	50	100	4
4	PCC 21CS44	Operating Systems		2	2	0		03	50	50	100	3
5	AEC 21BE45	Biology For Engineers	BT, CHE, PHY	2	0	0		02	50	50	100	2
6	PCC 21CSL46	Python Programming Laboratory	Any CS Board Department	0	0	2		03	50	50	100	1
7	HSMC 21KSK37/47	Sanskritika Kannada	HSMC	1	0	0		01	50	50	100	1
	HSMC 21KBK37/47	Balake Kannada										
	OR											
	HSMC 21CIP37/47	Constitution of India & Professional Ethics										
8	AEC 21CS48X/21C SL48X	Ability Enhancement Course- IV	TD and PSB: Concerned department	If offered as theory Course				01	50	50	100	1
				1	0	0						
				If offered as lab. course				02				
				0	0	2						
9	UHV 21UH49	Universal Human Values	Any Department	1	0	0		01	50	50	100	1
10	INT 21INT49	Inter/Intra Institutional Internship	Evaluation By the appropriate authorities	Completed during the intervening period of II and III semesters by students admitted to first year of BE./B.Tech and during the intervening period of III and IV semesters by Lateral entry students admitted to III semester.				3	100	--	100	2
Total									550	450	1000	22

Course prescribed to lateral entry Diploma holders admitted to III semester of Engineering programs

	NCMC 1 Additional 21MATDIP41	Mathematics - II Maths -- 100 0		02	02	--	--	--	100			
--	------------------------------------	---------------------------------	--	----	----	----	----	----	-----	--	--	--

Note: BSC: Basic Science Course, IPCC: Integrated Professional Core Course, PCC: Professional Core Course, AEC –Ability Enhancement Courses, HSMC: Humanity and Social Science and Management Courses, UHV- Universal Human Value Courses.

L –Lecture, T – Tutorial, P- Practical/ Drawing, S – Self Study Component, CIE: Continuous Internal Evaluation, SEE: Semester End Examination.

21KSK37/47 Sanskritika Kannada is for students who speak, read and write Kannada and 21KBK37/47 Balake Kannada is for non-Kannada speaking, reading, and writing students.

Integrated Professional Core Course (IPCC): Refers to Professional Theory Core Course Integrated with Practical's of the same course. Credit for IPCC can be 04 and its Teaching – Learning hours (L : T : P) can be considered as (3 : 0 : 2) or (2 : 2 : 2). The theory part of the IPCC shall be evaluated both by CIE and SEE. The practical part shall be evaluated by only CIE (no SEE). However, questions from practical part of IPCC shall be included in the SEE question paper. For more details the regulation governing the Degree of Bachelor of Engineering /Technology (BE/B.Tech.) 2021-22 may be referred.

Non – credit mandatory course (NCMC):

Additional Mathematics - II:

(1) Lateral entry Diploma holders admitted to III semester of B.E./B.Tech., shall attend the classes during the IV semester to complete all the formalities of the course and appear for the Continuous Internal Evaluation (CIE). In case, any student fails to register for the said course/fails to

secure the minimum 40 % of the prescribed CIE marks, he/she shall be deemed to have secured an F grade. In such a case, the student has to fulfil the course requirements during subsequent semester/s to earn the qualifying CIE marks. These courses are slated for CIE only and has no SEE.

(2) Additional Mathematics I and II shall not be considered for vertical progression as well as for the calculation of SGPA and CGPA, but completion of the courses shall be mandatory for the award of degree.

(3) Successful completion of the course Additional Mathematics II shall be indicated as satisfactory in the grade card. Non-completion of the courses Additional Mathematics II shall be indicated as Unsatisfactory.

Ability Enhancement Course - IV

21CSL481	Web Programming	21CSL483	R Programming
21CS482	Unix Shell Programming	21CS484	

Internship of 04 weeks during the intervening period of IV and V semesters; 21INT68Innovation/ Entrepreneurship/ Societalbased Internship.

(1)All the students shall have to undergo a mandatory internship of 04 weeks during the intervening period of IV and V semesters. The internship shall be slated for CIE only and will not have SEE. The letter grade earned through CIE shall be included in the VI semester grade card. The internship shall be considered as a head of passing and shall be considered for vertical progression and for the award of degree. Those, who do not take up / complete the internship shall be considered under F (fail) grade and shall have to complete during subsequently after satisfying the internship requirements.

(2)Innovation/ Entrepreneurship Internship shall be carried out at industry, State and Central Government /Non-government organizations (NGOs), micro, small and medium enterprise (MSME), Innovation centers or Incubation centers. Innovation need not be a single major breakthrough; it can also be a series of small or incremental changes. Innovation of any kind can also happen outside of the business world.

Entrepreneurship internships offers a chance to gain hands on experience in the world of entrepreneurship and helps to learn what it takes to run a small entrepreneurial business by performing intern duties with an established company. This experience can then be applied to future business endeavours. Start-ups and small companies are a preferred place to learn the business tack ticks for future entrepreneurs as learning how a small business operates will serve the intern well when he/she manages his/her own company. Entrepreneurship acts as a catalyst to open the minds to creativity and innovation. Entrepreneurship internship can be from several sectors, including technology, small and medium-sized, and the service sector.

(3) Societal or social internship.

Urbanization is increasing on a global scale; and yet, half the world's population still resides in rural areas and is devoid of many things that urban population enjoy. Rural internship, is a work-based activity in which students will have a chance to solve/reduce the problems of the rural place for better living.

As proposed under the AICTE rural internship programme, activities under Societal or social internship, particularly in rural areas, shall be considered for 40 points under AICTE activity point programme.

III Semester

TRANSFORM CALCULUS, FOURIER SERIES AND NUMERICAL TECHNIQUES			
Course Code:	21MAT31	CIE Marks	50
Teaching Hours/Week (L:T:P: S)	3:0:0:0	SEE Marks	50
Total Hours of Pedagogy	40	Total Marks	100
Credits	03	Exam Hours	03
Course Objectives:			
<p>CLO 1. To have an insight into solving ordinary differential equations by using Laplace transform techniques</p> <p>CLO 2. Learn to use the Fourier series to represent periodical physical phenomena in engineering analysis.</p> <p>CLO 3. To enable the students to study Fourier Transforms and concepts of infinite Fourier Sine and Cosine transforms and to learn the method of solving difference equations by the z-transform method.</p> <p>CLO 4. To develop the proficiency in solving ordinary and partial differential equations arising in engineering applications, using numerical methods</p>			
Teaching-Learning Process (General Instructions)			
<p>These are sample Strategies, which teachers can use to accelerate the attainment of the various course outcomes.</p> <ol style="list-style-type: none"> 1. Lecturer method (L) need not to be only traditional lecture method, but alternative effective teaching methods could be adopted to attain the outcomes. 2. Use of Video/Animation to explain functioning of various concepts. 3. Encourage collaborative (Group Learning) Learning in the class. 4. Ask at least three HOT (Higher order Thinking) questions in the class, which promotes critical thinking. 5. Adopt Problem Based Learning (PBL), which fosters students' Analytical skills, develop design thinking skills such as the ability to design, evaluate, generalize, and analyze information rather than simply recall it. 6. Introduce Topics in manifold representations. 7. Show the different ways to solve the same problem and encourage the students to come up with their own creative ways to solve them. 8. Discuss how every concept can be applied to the real world - and when that's possible, it helps improve the students' understanding. 			
Module-1			
<p>Definition $e^{at}f(t)$, $t^n f(t)$, $\frac{f(t)}{t}$ and Laplace transforms of elementary functions (statements only). Problems on Laplace transform of . Laplace transforms of Periodic functions (statement only) and unit-step function – problems.</p> <p>Inverse Laplace transforms definition and problems, Convolution theorem to find the inverse Laplace transforms (without Proof) and problems. Laplace transforms of derivatives, solution of differential equations.</p> <p>Self-study: Solution of simultaneous first-order differential equations.</p>			
Teaching-Learning Process		Chalk and talk method /	
Module-2			

Introduction to infinite series, convergence and divergence. Periodic functions, Dirichlet's condition. Fourier series of periodic functions with period 2π and arbitrary period. Half range Fourier series. Practical harmonic analysis.

Self-study: Convergence of series by D'Alembert's Ratio test and, Cauchy's root test

Teaching-Learning Process	Chalk and talk method / Powerpoint Presentation
----------------------------------	---

Module-3

Infinite Fourier transforms definition, Fourier sine and cosine transforms. Inverse Fourier transforms, Inverse Fourier cosine and sine transforms. Problems.

Difference equations, z-transform-definition, Standard z-transforms, Damping and shifting rules, Problems. Inverse z-transform and applications to solve difference equations.

Self-Study: Initial value and final value theorems, problems.

Teaching-Learning Process	Chalk and talk method / Powerpoint Presentation
----------------------------------	---

Module-4

Classifications of second-order partial differential equations, finite difference approximations to derivatives, Solution of Laplace's equation using standard five-point formula. Solution of heat equation by Schmidt explicit formula and Crank- Nicholson method, Solution of the Wave equation. Problems.

Self-Study: Solution of Poisson equations using standard five-point formula.

Teaching-Learning Process	Chalk and talk method / Powerpoint Presentation
----------------------------------	---

Module-5

Second-order differential equations - Runge-Kutta method and Milne's predictor and corrector method. (No derivations of formulae).

Calculus of Variations: Functionals, Euler's equation, Problems on extremals of functional. Geodesics on a plane, Variational problems.

Self-Study: Hanging chain problem

Teaching-Learning Process	Chalk and talk method / PowerPoint Presentation
----------------------------------	---

Course Outcomes (Course Skill Set)

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

- CO 1. To solve ordinary differential equations using Laplace transform.
- CO 2. Demonstrate Fourier series to study the behaviour of periodic functions and their applications in system communications, digital signal processing and field theory.
- CO 3. To use Fourier transforms to analyze problems involving continuous-time signals and to apply Z-Transform techniques to solve difference equations
- CO 4. To solve mathematical models represented by initial or boundary value problems involving partial differential equations
- CO 5. Determine the extremals of functionals using calculus of variations and solve problems arising in dynamics of rigid bodies and vibrational analysis.

Assessment Details (both CIE and SEE)

The weightage of Continuous Internal Evaluation (CIE) is 50% and for Semester End Exam (SEE) is 50%. The minimum passing mark for the CIE is 40% of the maximum marks (20 marks). A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to each subject/course if the student secures not less than 35% (18 Marks out of 50) in the semester-end examination (SEE), and a minimum of 40% (40 marks out of 100) in the sum total of the CIE (Continuous Internal Evaluation) and SEE (Semester End Examination) taken together **Continuous Internal Evaluation:**

Three Unit Tests each of **20 Marks (duration 01 hour)**

1. First test at the end of 5th week of the semester
2. Second test at the end of the 10th week of the semester
3. Third test at the end of the 15th week of the semester

Two assignments each of **10 Marks**

4. First assignment at the end of 4th week of the semester
5. Second assignment at the end of 9th week of the semester

Group discussion/Seminar/quiz any one of three suitably planned to attain the COs and POs for **20 Marks (duration 01 hours)**

6. At the end of the 13th week of the semester

The sum of three tests, two assignments, and quiz/seminar/group discussion will be out of 100 marks and will be **scaled down to 50 marks**

(to have less stressed CIE, the portion of the syllabus should not be common /repeated for any of the methods of the CIE. Each method of CIE should have a different syllabus portion of the course).

CIE methods /question paper has to be designed to attain the different levels of Bloom's taxonomy as per the outcome defined for the course.

Semester End Examination:

Theory SEE will be conducted by University as per the scheduled timetable, with common question papers for the subject (**duration 03 hours**)

1. The question paper will have ten questions. Each question is set for 20 marks. Marks scored shall be proportionally reduced to 50 marks.
2. There will be 2 questions from each module. Each of the two questions under a module (with a maximum of 3 sub-questions), **should have a mix of topics** under that module.

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

Suggested Learning Resources:**Textbooks**

1. B. S. Grewal: "Higher Engineering Mathematics", Khanna publishers, 44th Ed.2018
2. E. Kreyszig: "Advanced Engineering Mathematics", John Wiley & Sons, 10th Ed. (Reprint), 2016.

Reference Books:

1. V. Ramana: "Higher Engineering Mathematics" McGraw-Hill Education, 11th Ed.
2. Srimanta Pal & Subodh C. Bhunia: "Engineering Mathematics" Oxford University Press, 3rd Reprint, 2016.
3. N.P Bali and Manish Goyal: "A textbook of Engineering Mathematics" Laxmi Publications, Latest edition.
4. C. Ray Wylie, Louis C. Barrett: "Advanced Engineering Mathematics" McGraw - Hill Book Co.Newyork, Latest ed.
5. Gupta C.B, Sing S.R and Mukesh Kumar: "Engineering Mathematic for Semester I and II", Mc- Graw Hill Education(India) Pvt. Ltd 2015.
6. H.K.Dass and Er. Rajnish Verma: "Higher Engineering Mathematics" S.Chand Publication (2014).
7. James Stewart: "Calculus" Cengage publications, 7th edition, 4th Reprint 2019

Weblinks and Video Lectures (e-Resources):

1. [http://www.class-central.com/subject/math\(MOOCs\)](http://www.class-central.com/subject/math(MOOCs))
2. <http://academicearth.org/> 3. <http://www.bookstreet.in>.
4. VTU e-Shikshana Program
5. VTU EDUSAT Program

Activity Based Learning (Suggested Activities in Class)/ Practical Based learning

- Quizzes
- Assignments
- Seminars

III Semester**DATA STRUCTURES AND APPLICATIONS**

Course Code:	21CS32	CIE Marks	50
Teaching Hours/Week (L:T:P: S)	3:0:2:0	SEE Marks	50
Total Hours of Pedagogy	40 T + 20 P	Total Marks	100
Credits	04	Exam Hours	03

Course Objectives:

- CLO 1. Explain the fundamentals of data structures and their applications essential for implementing solutions to problems.
- CLO 2. Illustrate representation of data structures: Stack, Queues, Linked Lists, Trees and Graphs. CLO 3. Design and Develop Solutions to problems using Arrays, Structures, Stack, Queues, Linked Lists.
- CLO 4. Explore usage of Trees and Graph for application development.
- CLO 5. Apply the Hashing techniques in mapping key value pairs.

Teaching-Learning Process (General Instructions)

These are sample Strategies, which teachers can use to accelerate the attainment of the various course outcomes.

1. Lecturer method (L) need not to be only traditional lecture method, but alternative effective teaching methods could be adopted to attain the outcomes.
2. Use of Video/Animation to explain functioning of various concepts.
3. Encourage collaborative (Group Learning) Learning in the class.
4. Ask at least three HOT (Higher order Thinking) questions in the class, which promotes critical thinking.
5. Adopt Problem Based Learning (PBL), which fosters students' Analytical skills, develop design thinking skills such as the ability to design, evaluate, generalize, and analyze information rather than simply recall it.
6. Introduce Topics in manifold representations.
7. Show the different ways to solve the same problem and encourage the students to come up with their own creative ways to solve them.
8. Discuss how every concept can be applied to the real world - and when that's possible, it helps improve the students' understanding.

Module-1

Introduction: Data Structures, Classifications (Primitive & Non-Primitive), Data structure operations (Traversing, inserting, deleting, searching, and sorting). Review of Arrays. Structures: Array of structures Self-Referential Structures.

Dynamic Memory Allocation Functions. Representation of Linear Arrays in Memory, dynamically allocated arrays and Multidimensional Arrays.

Demonstration of representation of Polynomials and Sparse Matrices with arrays.

Textbook 1: Chapter 1: 1.2, Chapter 2: 2.2 - 2.7, Textbook 2: Chapter 1: 1.1 - 1.4, Chapter 3: 3.1 - 3.3, 3.5, 3.7, Chapter 4: 4.1 - 4.9, 4.14 Textbook 3: Chapter 1: 1.3

Laboratory Component:

1. Design, Develop and Implement a menu driven Program in C for the following Array Operations a. Creating an Array of N Integer Elements
b. Display of Array Elements with Suitable Headings c. Exit.
Support the program with functions for each of the above operations.
2. Design, Develop and Implement a menu driven Program in C for the following Array operations a. Inserting an Element (ELEM) at a given valid Position (POS)
b. Deleting an Element at a given valid Position POS)

- c. Display of Array Elements
 - d. Exit.
- Support the program with functions for each of the above operations.

Teaching-Learning Process

Problem based learning (Implementation of different programs to illustrate application of arrays and structures.

https://www.youtube.com/watch?v=3Xo6P_V-qns&t=201s

<https://ds2-iiith.vlabs.ac.in/exp/selection-sort/index.html>

<https://ds1-iiith.vlabs.ac.in/data-structures1/List%20of%20experiments.html>

Module-2

Stacks: Definition, Stack Operations, Array Representation of Stacks, Stacks using Dynamic Arrays. Different representation of expression. Stack Applications: Infix to postfix conversion, Infix to prefix conversion, evaluation of postfix expression, recursion.

Queues: Definition, Array Representation of Queues, Queue Operations, Circular Queues, Queues and Circular queues using Dynamic arrays, Dequeues, Priority Queues.

Textbook 1: Chapter 3: 3.1 -3.4, 3.6 Textbook 2: Chapter 6: 6.1 -6.4, 6.5, 6.7-6.13

Laboratory Component:

1. Design, Develop and Implement a menu driven Program in C for the following operations on STACK of Integers (Array Implementation of Stack with maximum size MAX)
 - a. *Push* an Element on to Stack
 - b. *Pop* an Element from Stack
 - c. Demonstrate *Overflow* and *Underflow* situations on Stack
 - d. Display the status of Stack
 - e. Exit

Support the program with appropriate functions for each of the above operations
2. Design, Develop and Implement a Program in C for the following Stack Applications
 - a. Evaluation of Suffix expression with single digit operands and operators: +, -, *, /, %, ^
 - b. Solving Tower of Hanoi problem with n disks

Teaching-Learning Process

Active Learning, Problem based learning
<https://nptel.ac.in/courses/106/102/106102064/>
<https://ds1-iiith.vlabs.ac.in/exp/stacks-queues/index.html>

Module-3

Linked Lists: Definition, classification of linked lists. Representation of different types of linked lists in Memory, Traversing, Insertion, Deletion, Searching, Sorting, and Concatenation Operations on Singly linked list, Doubly Linked lists, Circular linked lists, and header linked lists. Linked Stacks and Queues. Applications of Linked lists – Polynomials, Sparse matrix representation. Programming Examples.

Textbook 1: Chapter 4: 4.1 – 4.4, 4.5.2, 4.7, 4.8, Textbook 2: Chapter 5: 5.1 – 5.9

Laboratory Component:

1. Singly Linked List (SLL) of Integer Data
 - a. Create a SLL stack of N integer.
 - b. Display of SLL
 - c. Linear search. Create a SLL queue of N Students Data Concatenation of two SLL of integers.
2. Design, Develop and Implement a menu driven Program in C for the following operations on Doubly Linked List (DLL) of Professor Data with the fields: ID, Name, Branch, Area of specialization

- a. Create a DLL stack of N Professor's Data.
 - b. Create a DLL queue of N Professor's Data
- Display the status of DLL and count the number of nodes in it.

Teaching-Learning Process

MOOC, Active Learning, Problem solving based on linked lists.
<https://nptel.ac.in/courses/106/102/106102064/> <https://ds1-iiith.vlabs.ac.in/exp/linked-list/basics/overview.html> <https://ds1-iiith.vlabs.ac.in/List%20of%20experiments.html> <https://ds1-iiith.vlabs.ac.in/exp/linked-list/basics/overview.html> <https://ds1-iiith.vlabs.ac.in/List%20of%20experiments.html>

Module-4

Trees 1: Terminologies, Binary Trees, Properties of Binary trees, Array and linked Representation of Binary Trees, Binary Tree Traversals - Inorder, postorder, preorder; Threaded binary trees, Binary Search Trees – Definition, Insertion, Deletion, Traversal, and Searching operation on Binary search tree. Application of Trees-Evaluation of Expression.

Textbook 1: Chapter 5: 5.1 –5.5, 5.7; Textbook 2: Chapter 7: 7.1 – 7.9

Laboratory Component:

1. Given an array of elements, construct a complete binary tree from this array in level order fashion. That is, elements from left in the array will be filled in the tree level wise starting from level 0. Ex:
 Input : arr[] = {1, 2, 3, 4, 5, 6}
 Output : Root of the following tree


```

      1
      /\
      2 3
      /\  /\
      4 5 6
      
```
2. Design, Develop and Implement a menu driven Program in C for the following operations on Binary Search Tree (BST) of Integers
 - a. Create a BST of N Integers
 - b. Traverse the BST in Inorder, Preorder and Post Order

Teaching-Learning Process

Problem based learning
<http://www.nptelvideos.in/2012/11/data-structures-andalgorithms.html>
<https://ds1-iiith.vlabs.ac.in/exp/tree-traversal/index.html>
<https://ds1-iiith.vlabs.ac.in/exp/tree-traversal/depth-firsttraversal/dft-practice.html>

Module-5

Trees 2: AVL tree, Red-black tree, Splay tree, B-tree.

Graphs: Definitions, Terminologies, Matrix and Adjacency List Representation of Graphs, Traversal methods: Breadth First Search and Depth FirstSearch.

Hashing: Hash Table organizations, Hashing Functions, Static and Dynamic Hashing.

Textbook 1: Chapter 10:10.2, 10.3, 10.4, Textbook 2:7.10 – 7.12, 7.15 Chapter 11: 11.2, Textbook 1: Chapter 6 : 6.1–6.2, Chapter 8 : 8.1-8.3, Textbook 2: 8.1 – 8.3, 8.5, 8.7

Textbook 3: Chapter 15:15.1, 15.2,15.3, 15.4,15.5 and 15.7

--

Laboratory Component:

1. Design, Develop and implement a program in C for the following operations on Graph (G) of cities
 - a. Create a Graph of N cities using Adjacency Matrix.
 - b. Print all the nodes reachable from a given starting node in a diagraph using DFS/BFS method.
2. Design and develop a program in C that uses Hash Function $H:K \rightarrow L$ as $H(K)=K \bmod m$ (reminder method) and implement hashing technique to map a given key K to the address space L. Resolve the collision (if any) using linear probing.

Teaching-Learning Process

NPTL, MOOC etc. courses on trees and graphs.
<http://www.nptelvideos.in/2012/11/data-structures-andalgorithms.html>

Course Outcomes (Course Skill Set)

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

- CO 1. Identify different data structures and their applications.
- CO 2. Apply stack and queues in solving problems.
- CO 3. Demonstrate applications of linked list.
- CO 4. Explore the applications of trees and graphs to model and solve the real-world problem.
- CO 5. Make use of Hashing techniques and resolve collisions during mapping of key value pairs

Assessment Details (both CIE and SEE)

The weightage of Continuous Internal Evaluation (CIE) is 50% and for Semester End Exam (SEE) is 50%. The minimum passing mark for the CIE is 40% of the maximum marks (20 marks). A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to each subject/course if the student secures not less than 35% (18 Marks out of 50) in the semester-end examination (SEE), and a minimum of 40% (40 marks out of 100) in the sum total of the CIE (Continuous Internal Evaluation) and SEE (Semester End Examination) taken together **Continuous Internal Evaluation:**

Three Unit Tests each of **20 Marks (duration 01 hour)**

1. First test at the end of 5th week of the semester
2. Second test at the end of the 10th week of the semester
3. Third test at the end of the 15th week of the semester

Two assignments each of **10 Marks**

4. First assignment at the end of 4th week of the semester
5. Second assignment at the end of 9th week of the semester

Practical Sessions need to be assessed by appropriate rubrics and viva-voce method. This will contribute to **20 marks**.

- Rubrics for each Experiment taken average for all Lab components – 15 Marks.
- Viva-Voce– 5 Marks (more emphasized on demonstration topics)

The sum of three tests, two assignments, and practical sessions will be out of 100 marks and will be **scaled down to 50 marks**

(to have a less stressed CIE, the portion of the syllabus should not be common /repeated for any of the methods of the CIE. Each method of CIE should have a different syllabus portion of the course).

CIE methods /question paper has to be designed to attain the different levels of Bloom's taxonomy as per the outcome defined for the course.

Semester End Examination:

Theory SEE will be conducted by University as per the scheduled timetable, with common question papers for the subject (**duration 03 hours**)

1. The question paper will have ten questions. Each question is set for 20 marks. Marks scored shall be proportionally reduced to 50 Marks
2. There will be 2 questions from each module. Each of the two questions under a module (with a maximum of 3 sub-questions), **should have a mix of topics** under that module.

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

Suggested Learning Resources:**Textbooks:**

1. Ellis Horowitz and Sartaj Sahni, Fundamentals of Data Structures in C, 2nd Ed, Universities Press, 2014.
2. Seymour Lipschutz, Data Structures Schaum's Outlines, Revised 1st Ed, McGraw Hill, 2014.
3. Reema Thareja, Data Structures using C, 3rd Ed, Oxford press, 2012. **Reference Books:**
1. Gilberg and Forouzan, Data Structures: A Pseudo-code approach with C, 2nd Ed, Cengage Learning, 2014.
2. Jean-Paul Tremblay & Paul G. Sorenson, An Introduction to Data Structures with Applications, 2nd Ed, McGraw Hill, 2013
3. A M Tenenbaum, Data Structures using C, PHI, 1989
4. Robert Kruse, Data Structures and Program Design in C, 2nd Ed, PHI, 1996.

Weblinks and Video Lectures (e-Resources):

1. <http://elearning.vtu.ac.in/econtent/courses/video/CSE/06CS35.html>
2. <https://nptel.ac.in/courses/106/105/106105171/>
3. <http://www.nptelvideos.in/2012/11/data-structures-and-algorithms.html>

Activity Based Learning (Suggested Activities in Class)/ Practical Based learning

- Real world problem solving using group discussion.
- Back/Forward stacks on browsers.
- Undo/Redo stacks in Excel or Word.
- Linked list representation of real-world queues -Music player, image viewer

III Semester**ANALOG AND DIGITAL ELECTRONICS**

Course Code	21CS33	CIE Marks	50
Teaching Hours/Week (L:T:P: S)	3:0:2:0	SEE Marks	50
Total Hours of Pedagogy	40 T + 20 P	Total Marks	100
Credits	04	Exam Hours	03

Course Learning Objectives:

- CLO 1. Explain the use of photo electronics devices, 555 timer IC, Regulator ICs and uA741 CLO
2. Make use of simplifying techniques in the design of combinational circuits.
- CLO 3. Illustrate combinational and sequential digital circuits
- CLO 4. Demonstrate the use of flipflops and apply for registers
- CLO 5. Design and test counters, Analog-to-Digital and Digital-to-Analog conversion techniques.

Teaching-Learning Process (General Instructions)

These are sample Strategies, which teachers can use to accelerate the attainment of the various course outcomes.

1. Lecturer method (L) does not mean only traditional lecture method, but different type of teaching methods may be adopted to develop the outcomes.
2. Show Video/animation films to explain functioning of various concepts.
3. Encourage collaborative (Group Learning) Learning in the class.
4. Ask at least three HOT (Higher order Thinking) questions in the class, which promotes critical thinking.
5. Adopt Problem Based Learning (PBL), which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyze information rather than simply recall it.
6. Topics will be introduced in a multiple representation.
7. Show the different ways to solve the same problem and encourage the students to come up with their own creative ways to solve them.
8. Discuss how every concept can be applied to the real world - and when that's possible, it helps improve the students' understanding.

Module-1

BJT Biasing: Fixed bias, Collector to base Bias, voltage divider bias

Operational Amplifier Application Circuits: Peak Detector, Schmitt trigger, Active Filters, Non-Linear Amplifier, Relaxation Oscillator, Current-to-Voltage and Voltage-to-Current Converter, Regulated Power Supply Parameters, adjustable voltage regulator, D to A and A to D converter.

Textbook 1: Part A: Chapter 4 (Sections 4.2, 4.3, 4.4), Chapter 7 (Sections 7.4, 7.6 to 7.11), Chapter 8 (Sections 8.1 and 8.5), Chapter 9.

Laboratory Component:

1. Simulate BJT CE voltage divider biased voltage amplifier using any suitable circuit simulator.
2. Using ua 741 Opamp, design a 1 kHz Relaxation Oscillator with 50% duty cycle
3. Design an astable multivibrator circuit for three cases of duty cycle (50%, <50% and >50%) using NE 555 timer IC.
4. Using ua 741 opamp, design a window comparator for any given UTP and LTP.

Teaching-Learning Process

1. Demonstration of circuits using simulation.
2. Project work: Design a integrated power supply and function generator operating at audio frequency. Sine, square and triangular functions are to be generated.
3. Chalk and Board for numerical

Module-2

Karnaugh maps: minimum forms of switching functions, two and three variable Karnaugh maps, four variable Karnaugh maps, determination of minimum expressions using essential prime implicants, Quine-McClusky Method: determination of prime implicants, the prime implicant chart, Petricks method, simplification of incompletely specified functions, simplification using map-entered variables

Textbook 1: Part B: Chapter 5 (Sections 5.1 to 5.4) Chapter 6 (Sections 6.1 to 6.5)

Laboratory Component:

1. Given a 4-variable logic expression, simplify it using appropriate technique and implement the same using basic gates.

Teaching-Learning Process	<ol style="list-style-type: none"> 1. Chalk and Board for numerical 2. Laboratory Demonstration
Module-3	
<p>Combinational circuit design and simulation using gates: Review of Combinational circuit design, design of circuits with limited Gate Fan-in, Gate delays and Timing diagrams, Hazards in combinational Logic, simulation and testing of logic circuits</p> <p>Multiplexers, Decoders and Programmable Logic Devices: Multiplexers, three state buffers, decoders and encoders, Programmable Logic devices.</p> <p>Textbook 1: Part B: Chapter 8, Chapter 9 (Sections 9.1 to 9.6)</p>	
Laboratory Component:	
<ol style="list-style-type: none"> 1. Given a 4-variable logic expression, simplify it using appropriate technique and realize the simplified logic expression using 8:1 multiplexer IC. 2. Design and implement code converter I) Binary to Gray (II) Gray to Binary Code 	
Teaching-Learning Process	<ol style="list-style-type: none"> 1. Demonstration using simulator 2. Case study: Applications of Programmable Logic device 3. Chalk and Board for numerical
Module-4	
<p>Introduction to VHDL: VHDL description of combinational circuits, VHDL Models for multiplexers, VHDL Modules.</p> <p>Latches and Flip-Flops: Set Reset Latch, Gated Latches, Edge-Triggered D Flip Flop 3,SR Flip Flop, J K Flip Flop, T Flip Flop.</p> <p>Textbook 1: Part B: Chapter 10(Sections 10.1 to 10.3), Chapter 11 (Sections 11.1 to 11.7)</p>	
Laboratory Component:	
<ol style="list-style-type: none"> 1. Given a 4-variable logic expression, simplify it using appropriate technique and simulate the same in HDL simulator 2. Realize a J-K Master / Slave Flip-Flop using NAND gates and verify its truth table. And implement the same in HDL. 	
Teaching-Learning Process	<ol style="list-style-type: none"> 1. Demonstration using simulator 2. Case study: Arithmetic and Logic unit in VHDL 3. Chalk and Board for numerical
Module-5	
<p>Registers and Counters: Registers and Register Transfers, Parallel Adder with accumulator, shift registers, design of Binary counters, counters for other sequences, counter design using SR and J K Flip Flops.</p>	
<p>Textbook 1: Part B: Chapter 12 (Sections 12.1 to 12.5)</p>	

Laboratory Component:

1. Design and implement a mod-n ($n < 8$) synchronous up counter using J-K Flip-Flop ICs and demonstrate its working.
2. Design and implement an asynchronous counter using decade counter IC to count up from 0 to n ($n \leq 9$) and demonstrate on 7-segment display (using IC-7447)

Teaching-Learning Process

1. Demonstration using simulator
2. Project Work: Designing any counter, use LED / Sevensegment display to display the output
3. Chalk and Board for numerical

Course outcome (Course Skill Set)

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

- CO 1. Design and analyze application of analog circuits using photo devices, timer IC, power supply and regulator IC and op-amp.
- CO 2. Explain the basic principles of A/D and D/A conversion circuits and develop the same.
- CO 3. Simplify digital circuits using Karnaugh Map, and Quine-McClusky Methods
- CO 4. Explain Gates and flip flops and make us in designing different data processing circuits, registers and counters and compare the types. CO 5. Develop simple HDL programs

Assessment Details (both CIE and SEE)

The weightage of Continuous Internal Evaluation (CIE) is 50% and for Semester End Exam (SEE) is 50%. The minimum passing mark for the CIE is 40% of the maximum marks (20 marks). A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to each subject/course if the student secures not less than 35% (18 Marks out of 50) in the semester-end examination (SEE), and a minimum of 40% (40 marks out of 100) in the sum total of the CIE (Continuous Internal Evaluation) and SEE (Semester End Examination) taken together **Continuous Internal Evaluation:**

Three Unit Tests each of **20 Marks (duration 01 hour)**

1. First test at the end of 5th week of the semester
2. Second test at the end of the 10th week of the semester
3. Third test at the end of the 15th week of the semester

Two assignments each of **10 Marks**

4. First assignment at the end of 4th week of the semester
5. Second assignment at the end of 9th week of the semester

Practical Sessions need to be assessed by appropriate rubrics and viva-voce method. This will contribute to **20 marks**.

- Rubrics for each Experiment taken average for all Lab components – 15 Marks.
- Viva-Voce– 5 Marks (more emphasized on demonstration topics)

The sum of three tests, two assignments, and practical sessions will be out of 100 marks and will be **scaled down to 50 marks**

(to have a less stressed CIE, the portion of the syllabus should not be common /repeated for any of the methods of the CIE. Each method of CIE should have a different syllabus portion of the course).

CIE methods /question paper has to be designed to attain the different levels of Bloom's taxonomy as per the outcome defined for the course.

Semester End Examination:

Theory SEE will be conducted by University as per the scheduled timetable, with common question papers for the subject (**duration 03 hours**)

1. The question paper will have ten questions. Each question is set for 20 marks. Marks scored shall be proportionally reduced to 50 marks
2. There will be 2 questions from each module. Each of the two questions under a module (with a maximum of 3 sub-questions), **should have a mix of topics** under that module.

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

Suggested Learning Resources:

Textbooks

1. Charles H Roth and Larry L Kinney and Raghunandan G H Analog and Digital Electronics, Cengage Learning, 2019

Reference 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.
3. M. Morris Mani, Digital Design, 4th Edition, Pearson Prentice Hall, 2008.
4. David A. Bell, Electronic Devices and Circuits, 5th Edition, Oxford University Press, 2008

Weblinks and Video Lectures (e-Resources):

1. Analog Electronic Circuits: <https://nptel.ac.in/courses/108/102/108102112/>
2. Digital Electronic Circuits: <https://nptel.ac.in/courses/108/105/108105132/>
3. Analog Electronics Lab: <http://vlabs.iitkgp.ac.in/be/>
4. Digital Electronics Lab: <http://vlabs.iitkgp.ac.in/dec>

Activity Based Learning (Suggested Activities in Class)/ Practical Based learning

1. Real world problem solving - applying the design concepts of oscillator; amplifier; switch, Digital circuits using Opamps, 555 timer, transistor; Digital ICs and design a application like tone generator; temperature sensor; digital clock, dancing lights etc.

III Semester

COMPUTER ORGANIZATION AND ARCHITECTURE

Course Code	21CS34	CIE Marks	50
Teaching Hours/Week (L:T:P: S)	3:0:0:0	SEE Marks	50
Total Hours of Pedagogy	40	Total Marks	100
Credits	03	Exam Hours	03

Course Learning Objectives

- CLO 1. Understand the organization and architecture of computer systems, their structure and operation
- CLO 2. Illustrate the concept of machine instructions and programs
- CLO 3. Demonstrate different ways of communicating with I/O devices
- CLO 4. Describe different types memory devices and their functions
- CLO 5. Explain arithmetic and logical operations with different data types
- CLO 6. Demonstrate processing unit with parallel processing and pipeline architecture

Teaching-Learning Process (General Instructions)

These are sample Strategies, which teachers can use to accelerate the attainment of the various course outcomes.

1. Lecturer method (L) need not to be only a traditional lecture method, but alternative effective teaching methods could be adopted to attain the outcomes.
2. Use of Video/Animation to explain functioning of various concepts.
3. Encourage collaborative (Group Learning) Learning in the class.
4. Ask at least three HOT (Higher order Thinking) questions in the class, which promotes critical thinking.
5. Adopt Problem Based Learning (PBL), which fosters students' Analytical skills, develop design thinking skills such as the ability to design, evaluate, generalize, and analyze information rather than simply recall it.
6. Introduce Topics in manifold representations.
7. Show the different ways to solve the same problem with different circuits/logic and encourage the students to come up with their own creative ways to solve them.
8. Discuss how every concept can be applied to the real world - and when that's possible, it helps improve the students' understanding.

Module-1

Basic Structure of Computers: Basic Operational Concepts, Bus Structures, Performance – Processor Clock, Basic Performance Equation, Clock Rate, Performance Measurement.

Machine Instructions and Programs: Memory Location and Addresses, Memory Operations, Instructions and Instruction Sequencing, Addressing Modes

Textbook 1: Chapter1 – 1.3, 1.4, 1.6 (1.6.1-1.6.4, 1.6.7), Chapter2 – 2.2 to 2.5

Teaching-Learning Process	Chalk and board, Active Learning, Problem based learning
----------------------------------	--

Module-2

Input/Output Organization: Accessing I/O Devices, Interrupts – Interrupt Hardware, Direct Memory Access, Buses, Interface Circuits

Textbook 1: Chapter4 – 4.1, 4.2, 4.4, 4.5, 4.6

Teaching-Learning Process	Chalk and board, Active Learning, Demonstration
----------------------------------	---

Module-3

Memory System: Basic Concepts, Semiconductor RAM Memories, Read Only Memories, Speed, Size, and Cost, Cache Memories – Mapping Functions, Virtual memories

Textbook 1: Chapter 5 – 5.1 to 5.4, 5.5 (5.5.1, 5.5.2)

Teaching-Learning Process	Chalk and board, Problem based learning, Demonstration
----------------------------------	--

Module-4

Arithmetic: Numbers, Arithmetic Operations and Characters, Addition and Subtraction of Signed Numbers, Design of Fast Adders, Multiplication of Positive Numbers

Basic Processing Unit: Fundamental Concepts, Execution of a Complete Instruction, Hardwired control, Microprogrammed control

Textbook 1: Chapter2-2.1, Chapter6 – 6.1 to 6.3

Textbook 1: Chapter7 – 7.1, 7.2,7.4, 7.5

Teaching-Learning Process	Chalk& board, Problem based learning
----------------------------------	--------------------------------------

Module-5

Pipeline and Vector Processing: Parallel Processing, Pipelining, Arithmetic Pipeline, Instruction Pipeline, Vector Processing, Array Processors

Textbook 2: Chapter 9 – 9.1, 9.2, 9.3, 9.4, 9.6, 9.7

Teaching-Learning Process | Chalk and board, MOOC

Course Outcomes

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

- CO 1. Explain the organization and architecture of computer systems with machine instructions and programs
- CO 2. Analyze the input/output devices communicating with computer system
- CO 3. Demonstrate the functions of different types of memory devices
- CO 4. Apply different data types on simple arithmetic and logical unit
- CO 5. Analyze the functions of basic processing unit, Parallel processing and pipelining

Assessment Details (both CIE and SEE)

The weightage of Continuous Internal Evaluation (CIE) is 50% and for Semester End Exam (SEE) is 50%. The minimum passing mark for the CIE is 40% of the maximum marks (20 marks). A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to each subject/course if the student secures not less than 35% (18 Marks out of 50) in the semester-end examination (SEE), and a minimum of 40% (40 marks out of 100) in the sum total of the CIE (Continuous Internal Evaluation) and SEE (Semester End Examination) taken together **Continuous Internal Evaluation:**

Three Unit Tests each of **20 Marks (duration 01 hour)**

1. First test at the end of 5th week of the semester
2. Second test at the end of the 10th week of the semester
3. Third test at the end of the 15th week of the semester

Two assignments each of **10 Marks**

4. First assignment at the end of 4th week of the semester
5. Second assignment at the end of 9th week of the semester

Group discussion/Seminar/quiz any one of three suitably planned to attain the COs and POs for **20 Marks (duration 01 hours)**

6. At the end of the 13th week of the semester

The sum of three tests, two assignments, and quiz/seminar/group discussion will be out of 100 marks and will be **scaled down to 50 marks**

(to have less stressed CIE, the portion of the syllabus should not be common /repeated for any of the methods of the CIE. Each method of CIE should have a different syllabus portion of the course).

CIE methods /question paper has to be designed to attain the different levels of Bloom's taxonomy as per the outcome defined for the course.

Semester End Examination:

Theory SEE will be conducted by University as per the scheduled timetable, with common question papers for the subject (**duration 03 hours**)

1. The question paper will have ten questions. Each question is set for 20 marks. Marks scored shall be proportionally reduced to 50 marks
2. There will be 2 questions from each module. Each of the two questions under a module (with a maximum of 3 sub-questions), **should have a mix of topics** under that module.

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

Textbooks

1. Carl Hamacher, Zvonko Vranesic, Safwat Zaky, Computer Organization, 5th Edition, Tata McGraw Hill
2. M. Morris Mano, Computer System Architecture, PHI, 3rd Edition **Reference:**
 1. William Stallings: Computer Organization & Architecture, 9th Edition, Pearson

Weblinks and Video Lectures (e-Resources):

1. <https://nptel.ac.in/courses/106/103/106103068/>
2. <https://nptel.ac.in/content/storage2/courses/106103068/pdf/coa.pdf>
3. <https://nptel.ac.in/courses/106/105/106105163/> 4. <https://nptel.ac.in/courses/106/106/106106092/>
5. <https://nptel.ac.in/courses/106/106/106106166/>
6. <http://www.nptelvideos.in/2012/11/computer-organization.html>

Activity Based Learning (Suggested Activities in Class)/ Practical Based learning

- Discussion and literature survey on real world use cases
- Quizzes

III Semester

OBJECT ORIENTED PROGRAMMING WITH JAVA LABORATORY			
Course Code	21CSL35	CIE Marks	50
Teaching Hours/Week (L:T:P: S)	0:0:2:0	SEE Marks	50
Total Hours of Pedagogy	24	Total Marks	100
Credits	1	Exam Hours	03
Course Objectives:			
CLO 1. Demonstrate the use of Eclipse/Netbeans IDE to create Java Applications.			
CLO 2. Using java programming to develop programs for solving real-world problems. CLO			
3. Reinforce the understanding of basic object-oriented programming concepts.			
Note: two hours tutorial is suggested for each laboratory sessions.			
Prerequisite			
<ul style="list-style-type: none"> • Students should be familiarized about java installation and setting the java environment. • Usage of IDEs like Eclipse/Netbeans should be introduced. 			
Sl. No.	<i>PART A - List of problems for which student should develop program and execute in the Laboratory</i>		
1	Aim: Introduce the java fundamentals, data types, operators in java Program: Write a java program that prints all real solutions to the quadratic equation $ax^2+bx+c=0$. Read in a, b, c and use the quadratic formula.		
2	Aim: Demonstrating creation of java classes, objects, constructors, declaration and initialization of variables. Program: Create a Java class called Student with the following details as variables within it. USN Name Branch Phone Write a Java program to create n Student objects and print the USN, Name, Branch, and Phone of these objects with suitable headings.		
3	Aim: Discuss the various Decision-making statements, loop constructs in java Program: A. Write a program to check prime number B. Write a program for Arithmetic calculator using switch case menu		
4	Aim: Demonstrate the core object-oriented concept of Inheritance, polymorphism Design a super class called Staff with details as StaffId, Name, Phone, Salary. Extend this class by writing three subclasses namely Teaching (domain, publications), Technical (skills), and Contract (period). Write a Java program to read and display at least 3 staff objects of all three categories.		
5	Aim: Introduce concepts of method overloading, constructor overloading, overriding. Program: Write a java program demonstrating Method overloading and Constructor overloading.		

6	Aim: Introduce the concept of Abstraction, packages. Program: Develop a java application to implement currency converter (Dollar to INR, EURO to INR, Yen to INR and vice versa), distance converter (meter to KM, miles to KM and vice versa), time converter (hours to minutes, seconds and vice versa) using packages.
7	Aim: Introduction to abstract classes, abstract methods, and Interface in java

	Program: Write a program to generate the resume. Create 2 Java classes Teacher (data: personal information, qualification, experience, achievements) and Student (data: personal information, result, discipline) which implements the java interface Resume with the method biodata().
8	Aim: Demonstrate creation of threads using Thread class and Runnable interface, multithreaded programming. Program: Write a Java program that implements a multi-thread application that has three threads. First thread generates a random integer for every 1 second; second thread computes the square of the number and prints; third thread will print the value of cube of the number.
9	Aim: Introduce java Collections. Program: Write a program to perform string operations using ArrayList. Write functions for the following a. Append - add at end b. Insert – add at particular index c. Search d. List all string starts with given letter.
10	Aim: Exception handling in java, introduction to throwable class, throw, throws, finally. Program: Write a Java program to read two integers a and b. Compute a/b and print, when b is not zero. Raise an exception when b is equal to zero.
11	Aim: Introduce File operations in java. Program: Write a java program that reads a file name from the user, displays information about whether the file exists, whether the file is readable, or writable, the type of file and the length of the file in bytes
12	Aim: Introduce java Applet, awt, swings. Programs: Develop an applet that displays a simple message in center of the screen. Develop a simple calculator using Swings.
PART B – Practical Based Learning	
01	A problem statement for each batch is to be generated in consultation with the co-examiner and student should develop an algorithm, program and execute the program for the given problem with appropriate outputs.

Course Outcome (Course Skill Set)

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

- CO 1. Use Eclipse/NetBeans IDE to design, develop, debug Java Projects.
- CO 2. Analyze the necessity for Object Oriented Programming paradigm over structured programming and become familiar with the fundamental concepts in OOP.
- CO 3. Demonstrate the ability to design and develop java programs, analyze, and interpret objectoriented data and document results.
- CO 4. Apply the concepts of multiprogramming, exception/event handling, abstraction to develop robust programs.

CO 5. Develop user friendly applications using File I/O and GUI concepts.

Assessment Details (both CIE and SEE)

The weightage of Continuous Internal Evaluation (CIE) is 50% and for Semester End Exam (SEE) is 50%. The minimum passing mark for the CIE is 40% of the maximum marks (20 marks). A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to each course. The student has to secure not less than 35% (18 Marks out of 50) in the semester-end examination (SEE).

Continuous Internal Evaluation (CIE):

CIE marks for the practical course is **50 Marks**.

The split-up of CIE marks for record/ journal and test are in the ratio **60:40**.

- Each experiment to be evaluated for conduction with observation sheet and record write-up. Rubrics for the evaluation of the journal/write-up for hardware/software experiments designed by the faculty who is handling the laboratory session and is made known to students at the beginning of the practical session.
- Record should contain all the specified experiments in the syllabus and each experiment write-up will be evaluated for 10 marks.
- Total marks scored by the students are scaled down to 30 marks (60% of maximum marks).
- Weightage to be given for neatness and submission of record/write-up on time.
- Department shall conduct 02 tests for 100 marks, the first test shall be conducted after the 8th week of the semester and the second test shall be conducted after the 14th week of the semester.
- In each test, test write-up, conduction of experiment, acceptable result, and procedural knowledge will carry a weightage of 60% and the rest 40% for viva-voce.
- The suitable rubrics can be designed to evaluate each student's performance and learning ability. Rubrics suggested in Annexure-II of Regulation book
- The average of 02 tests is scaled down to **20 marks** (40% of the maximum marks).

The Sum of scaled-down marks scored in the report write-up/journal and average marks of two tests is the total CIE marks scored by the student.

Semester End Evaluation (SEE):

- SEE marks for the practical course is 50 Marks.
- SEE shall be conducted jointly by the two examiners of the same institute, examiners are appointed by the University
- All laboratory experiments are to be included for practical examination.
- (Rubrics) Breakup of marks and the instructions printed on the cover page of the answer script to be strictly adhered to by the examiners. **OR** based on the course requirement evaluation rubrics shall be decided jointly by examiners.
- Students can pick one question (experiment) from the questions lot prepared by the internal /external examiners jointly.
- Evaluation of test write-up/ conduction procedure and result/viva will be conducted jointly by examiners.
- General rubrics suggested for SEE are mentioned here, writeup-20%, Conduction procedure and result in -60%, Viva-voce 20% of maximum marks. SEE for practical shall be evaluated for 100 marks and scored marks shall be scaled down to 50 marks (however, based on course type, rubrics shall be decided by the examiners)
- *Students can pick one experiment from the questions lot of PART A with equal choice to all the students in a batch. For PART B examiners should frame a question for each batch, student should develop an algorithm, program, execute and demonstrate the results with appropriate output for the given problem.*
- *Weightage of marks for PART A is 80% and for PART B is 20%. General rubrics suggested to be followed for part A and part B.*
- Change of experiment is allowed only once and Marks allotted to the procedure part to be made zero (Not allowed for Part B).
- The duration of SEE is 03 hours
- Rubrics suggested in Annexure-II of Regulation book

Suggested Learning Resources:

1. E Balagurusamy, Programming with Java, Graw Hill, 6th Edition, 2019.
2. Herbert Schildt, C: Java the Complete Reference, McGraw Hill, 11th Edition, 2020

III Semester

MASTERING OFFICE (Practical based)			
Course Code	21CSL381	CIE Marks	50
Teaching Hours/Week (L:T:P: S)	1:0:0:0	SEE Marks	50
Total Hours of Pedagogy	12T + 12P	Total Marks	100
Credits	01	Exam Hours	02
Course Objectives:			
CLO 1. Understand the basics of computers and prepare documents and small presentations.			
CLO 2. Attain the knowledge about spreadsheet/worksheet with various options.			
CLO 3. Create simple presentations using templates various options available.			
CLO 4. Demonstrate the ability to apply application software in an office environment.			
CLO 5. Use MS Office to create projects, applications.			

Teaching-Learning Process (General Instructions)

These are sample Strategies, which teachers can use to accelerate the attainment of the various course outcomes.

1. Lecturer method (L) need not to be only traditional lecture method, but alternative effective teaching methods could be adopted to attain the outcomes.
2. Use of Video/Animation to explain functioning of various concepts.
3. Encourage collaborative (Group Learning) Learning in the class.
4. Ask at least three HOT (Higher order Thinking) questions in the class, which promotes critical thinking.
5. Adopt Problem Based Learning (PBL), which fosters students' Analytical skills, develop design thinking skills such as the ability to design, evaluate, generalize, and analyze information rather than simply recall it.
6. Introduce Topics in manifold representations.
7. Show the different ways to solve the same problem with different circuits/logic and encourage the students to come up with their own creative ways to solve them.
8. Discuss how every concept can be applied to the real world - and when that's possible, it helps improve the students' understanding.

Module-1

MS-Word -Working with Files, Text – Formatting, Moving, copying and pasting text, Styles – Lists – Bulleted and numbered lists, Nested lists, Formatting lists. Table Manipulations. Graphics – Adding clip Art, add an image from a file, editing graphics, Page formatting - Header and footers, page numbers, Protect the Document, Mail Merge, Macros – Creating & Saving web pages, Hyperlinks.

Textbook 1: Chapter 2

Teaching-Learning Process	Chalk and board, Active Learning, practical based learning
----------------------------------	--

Module-2

MS-Excel- Modifying a Worksheet – Moving through cells, adding worksheets, rows and columns, Resizing rows and columns, selecting cells, Moving and copying cells, freezing panes - Macros – recording and running. Linking worksheets - Sorting and Filling, Alternating text and numbers with Auto fill, Auto filling functions. Graphics – Adding clip art, add an image from a file, Charts – Using chart Wizard, Copy a chart to Microsoft Word.

Textbook 1: Chapter 3

Teaching-Learning Process	Active Learning, Demonstration, presentation,
----------------------------------	---

Module-3

MS-Power Point -Create a Presentation from a template- Working with Slides – Insert a new slide, applying a design template, changing slide layouts – Resizing a text box, Text box properties, delete a text box - Video and Audio effects, Color Schemes & Backgrounds Adding clip art, adding an image from a file, Save as a web page.

Textbook 1: Chapter 5

Teaching-Learning Process	Demonstration, presentation preparation for case studies
----------------------------------	--

Module-4

MS-Access - Using Access database wizard, pages and projects. Creating Tables – Create a Table in design view. Datasheet Records – Adding, Editing, deleting records, Adding and deleting columns Resizing rows and columns, finding data in a table & replacing, Print a datasheet. Queries - MS-Access.

Textbook 1: Chapter 4

Teaching-Learning Process	Chalk& board, Practical based learning.
Module-5	
Microsoft Outlook- Introduction, Starting Microsoft Outlook, Outlook Today, Different Views In Outlook, Outlook Data Files	
Textbook 1: Chapter 7	
Teaching-Learning Process	Chalk and board, MOOC
<p>Course Outcomes (Course Skill Set): At the end of the course the student will be able to:</p> <ul style="list-style-type: none"> CO 1. Know the basics of computers and prepare documents, spreadsheets, make small presentations with audio, video and graphs and would be acquainted with internet. CO 2. Create, edit, save and print documents with list tables, header, footer, graphic, spellchecker, mail merge and grammar checker CO 3. Attain the knowledge about spreadsheet with formula, macros spell checker etc. CO 4. Demonstrate the ability to apply application software in an office environment. CO 5. Use Google Suite for office data management tasks 	
<p>Assessment Details (both CIE and SEE)</p> <p>The weightage of Continuous Internal Evaluation (CIE) is 50% and for Semester End Exam (SEE) is 50%. The minimum passing mark for the CIE is 40% of the maximum marks (20 marks). A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to each course. The student has to secure not less than 35% (18 Marks out of 50) in the semester-end examination (SEE).</p> <p>Continuous Internal Evaluation (CIE): NOTE: List of experiments to be prepared by the faculty based on the syllabus mentioned above CIE marks for the practical course is 50 Marks.</p> <p>The split-up of CIE marks for record/ journal and test are in the ratio 60:40.</p> <ul style="list-style-type: none"> • Each experiment to be evaluated for conduction with observation sheet and record write-up. Rubrics for the evaluation of the journal/write-up for hardware/software experiments designed by the faculty who is handling the laboratory session and is made known to students at the beginning of the practical session. • Record should contain all the specified experiments in the syllabus and each experiment write-up will be evaluated for 10 marks. • Total marks scored by the students are scaled down to 30 marks (60% of maximum marks). • Weightage to be given for neatness and submission of record/write-up on time. • Department shall conduct 02 tests for 100 marks, the first test shall be conducted after the 8th week of the semester and the second test shall be conducted after the 14th week of the semester. • In each test, test write-up, conduction of experiment, acceptable result, and procedural knowledge will carry a weightage of 60% and the rest 40% for viva-voce. • The suitable rubrics can be designed to evaluate each student's performance and learning ability. Rubrics suggested in Annexure-II of Regulation book • The average of 02 tests is scaled down to 20 marks (40% of the maximum marks). <p>The Sum of scaled-down marks scored in the report write-up/journal and average marks of two tests is the total CIE marks scored by the student.</p>	
Semester End Evaluation (SEE):	

- SEE marks for the practical course is 50 Marks.
- SEE shall be conducted jointly by the two examiners of the same institute, examiners are appointed by the University
- All laboratory experiments are to be included for practical examination.
- (Rubrics) Breakup of marks and the instructions printed on the cover page of the answer script to be strictly adhered to by the examiners. **OR** based on the course requirement evaluation rubrics shall be decided jointly by examiners.
- Students can pick one question (experiment) from the questions lot prepared by the internal /external examiners jointly.
- Evaluation of test write-up/ conduction procedure and result/viva will be conducted jointly by examiners.
- General rubrics suggested for SEE are mentioned here, writeup-20%, Conduction procedure and result in -60%, Viva-voce 20% of maximum marks. SEE for practical shall be evaluated for 100 marks and scored marks shall be scaled down to 50 marks (however, based on course type, rubrics shall be decided by the examiners)
- The duration of SEE is 02 hours

Rubrics suggested in Annexure-II of Regulation book

Weblinks and Video Lectures (e-Resources):

1. <https://youtu.be/9VRmgC2GRFE>
2. <https://youtu.be/rJPWi5x0g3I>
3. <https://youtu.be/tcj2BhhCMN4>
4. <https://youtu.be/ubmwp8kbfPc>
5. <https://youtu.be/i6eNvfQ8fTw>
6. <http://office.microsoft.com/en-us/training/CR010047968.aspx>
7. <https://gsuite.google.com/learning-center>
8. <http://spoken-tutorial.org>

Activity Based Learning (Suggested Activities in Class)/ Practical Based learning ☐

Real world problem solving using group discussion.

- ☐ Real world examples of Windows Framework.

III Semester

PROGRAMMING IN C++			
Course Code	21CS382	CIE Marks	50
Teaching Hours/Week (L:T:P: S)	1:0:0:0	SEE Marks	50
Total Hours of Pedagogy	12	Total Marks	100
Credits	01	Exam Hours	01
Course Objectives:			
CLO 1. Understanding about object oriented programming and Gain knowledge about the capability to store information together in an object.			
CLO 2. Understand the capability of a class to rely upon another class and functions.			
CLO 3. Understand about constructors which are special type of functions.			
CLO 4. Create and process data in files using file I/O functions			
CLO 5. Use the generic programming features of C++ including Exception handling.			

Teaching-Learning Process (General Instructions)

These are sample Strategies, which teachers can use to accelerate the attainment of the various course outcomes.

1. Lecturer method (L) need not to be only a traditional lecture method, but alternative effective teaching methods could be adopted to attain the outcomes.
2. Use of Video/Animation to explain functioning of various concepts.
3. Encourage collaborative (Group Learning) Learning in the class.
4. Ask at least three HOT (Higher order Thinking) questions in the class, which promotes critical thinking.
5. Adopt Problem Based Learning (PBL), which fosters students' Analytical skills, develop design thinking skills such as the ability to design, evaluate, generalize, and analyze information rather than simply recall it.
6. Introduce Topics in manifold representations.
7. Show the different ways to solve the same problem with different circuits/logic and encourage the students to come up with their own creative ways to solve them.
8. Discuss how every concept can be applied to the real world - and when that's possible, it helps improve the students' understanding.

Module-1

Introduction to Object Oriented Programming: Computer programming background- C++ overview First C++ Program -Basic C++ syntax, Object Oriented Programming: What is an object, Classes, methods and messages, abstraction and encapsulation, inheritance, abstract classes, polymorphism.

Textbook 1: Chapter 1(1.1 to 1.8)

Teaching-Learning Process

Chalk and board, Active Learning, practical based learning

Module-2

Functions in C++: Tokens – Keywords – Identifiers and constants – Operators in C++ – Scope resolution operator – Expressions and their types – Special assignment expressions – Function prototyping – Call by reference – Return by reference – Inline functions -Default arguments – Function overloading.

Textbook 2: Chapter 3(3.2,3.3,3.4,3.13,3.14,3.19, 3.20) , chapter 4(4.3,4.4,4.5,4.6,4.7,4.9)

Teaching-Learning Process

Chalk and board, Active Learning, Demonstration, presentation, problem solving

Module-3

Inheritance & Polymorphism: Derived class Constructors, destructors-Types of Inheritance- Defining Derived classes, Single Inheritance, Multiple, Hierarchical Inheritance, Hybrid Inheritance.

Textbook 2: Chapter 6 (6.2,6.11) chapter 8 (8.1 to,8.8)

Teaching-Learning Process

Chalk and board, Demonstration, problem solving

Module-4

I/O Streams: C++ Class Hierarchy- File Stream-Text File Handling- Binary File Handling during file operations.

Textbook 1: Chapter 12(12.5) , Chapter 13 (13.6,13.7)

Teaching-Learning Process

Chalk and board, Practical based learning, practical's

Module-5

Exception Handling: Introduction to Exception - Benefits of Exception handling- Try and catch block-Throw statement- Pre-defined exceptions in C++ .

Textbook 2: Chapter 13 (13.2 to13.6)

Teaching-Learning Process

Chalk and board, MOOC

Course Outcomes (Course Skill Set):

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

- CO 1. Able to understand and design the solution to a problem using object-oriented programming concepts.
- CO 2. Able to reuse the code with extensible Class types, User-defined operators and function Overloading.
- CO 3. Achieve code reusability and extensibility by means of Inheritance and Polymorphism
- CO 4. Identify and explore the Performance analysis of I/O Streams.
- CO 5. Implement the features of C++ including templates, exceptions and file handling for providing programmed solutions to complex problems.

Assessment Details (both CIE and SEE)

The weightage of Continuous Internal Evaluation (CIE) is 50% and for Semester End Exam (SEE) is 50%. The minimum passing mark for the CIE is 40% of the maximum marks (20 marks). A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to each subject/course if the student secures not less than 35% (18 Marks out of 50) in the semester-end examination (SEE), and a minimum of 40% (40 marks out of 100) in the sum total of the CIE (Continuous Internal Evaluation) and SEE (Semester End Examination) taken together

Continuous Internal Evaluation:

Three Unit Tests each of **20 Marks (duration 01 hour)**

- 1. First test at the end of 5th week of the semester
- 2. Second test at the end of the 10th week of the semester
- 3. Third test at the end of the 15th week of the semester

Two assignments each of **10 Marks**

- 4. First assignment at the end of 4th week of the semester
- 5. Second assignment at the end of 9th week of the semester

Group discussion/Seminar/quiz any one of three suitably planned to attain the COs and POs for **20 Marks (duration 01 hours)**

- 6. At the end of the 13th week of the semester

The sum of three tests, two assignments, and quiz/seminar/group discussion will be out of 100 marks and will be **scaled down to 50 marks**

(to have less stressed CIE, the portion of the syllabus should not be common /repeated for any of the methods of the CIE. Each method of CIE should have a different syllabus portion of the course).

CIE methods /question paper has to be designed to attain the different levels of Bloom's taxonomy as per the outcome defined for the course.

Semester End Examination:

Theory SEE will be conducted by University as per the scheduled timetable, with common question papers for the subject (**duration 01 hours**)

SEE paper will be set for 50 questions of each of 01 marks. The pattern of the question paper is MCQ. The time allotted for SEE is 01 hours

Textbooks

1. Bhushan Trivedi, "Programming with ANSI C++", Oxford Press, Second Edition, 2012.
2. Balagurusamy E, Object Oriented Programming with C++, Tata McGraw Hill Education Pvt.Ltd , Fourth Edition 2010.

Reference Books

1. Bhave , " Object Oriented Programming With C++", Pearson Education , 2004.
2. Ray Lischner, "Exploring C++ : The programmer's introduction to C++" , apress, 2010
3. Bhave , " Object Oriented Programming With C++", Pearson Education , 2004

Weblinks and Video Lectures (e-Resources):

1. Basics of C++ - <https://www.youtube.com/watch?v=BCIS40yzssA>
2. Functions of C++ - <https://www.youtube.com/watch?v=p8ehAjZWjPw> **Tutorial Link:**
1. https://www.w3schools.com/cpp/cpp_intro.asp
2. <https://www.edx.org/course/introduction-to-c-3>

Activity Based Learning (Suggested Activities in Class)/ Practical Based learning

- **Demonstration of simple projects**

IV Semester

--

--	--

--

--

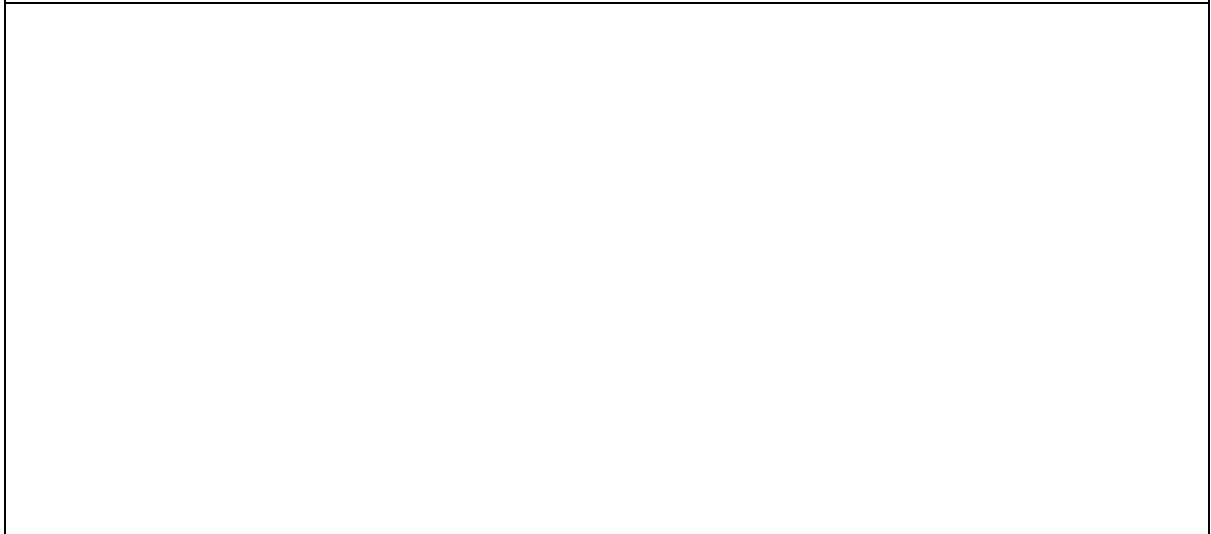
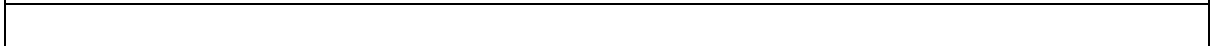
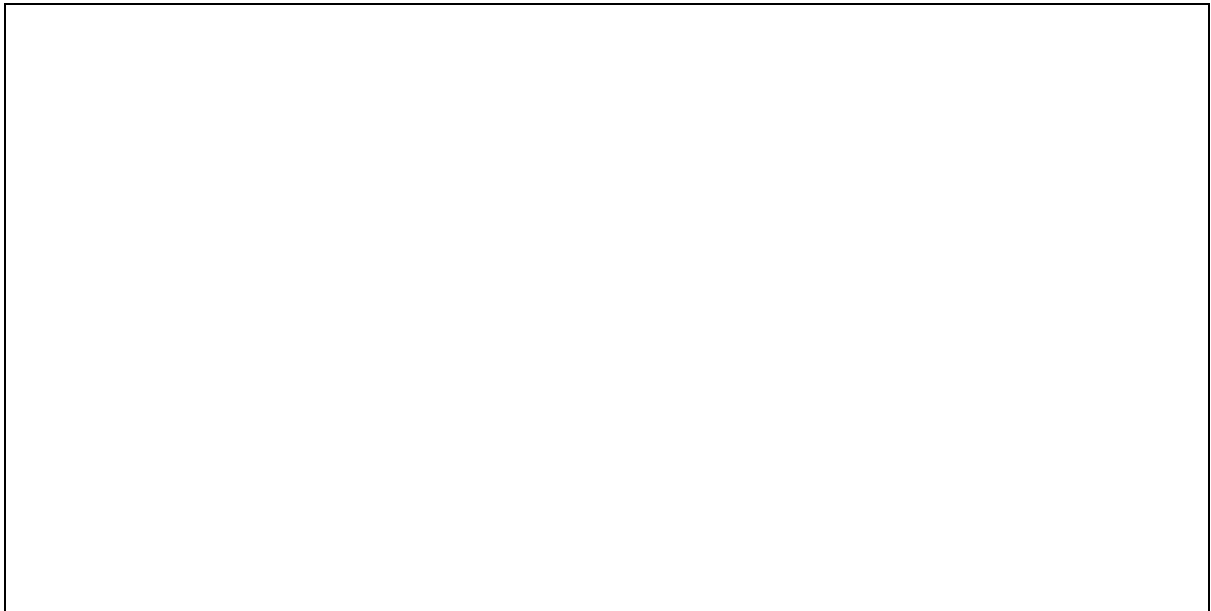
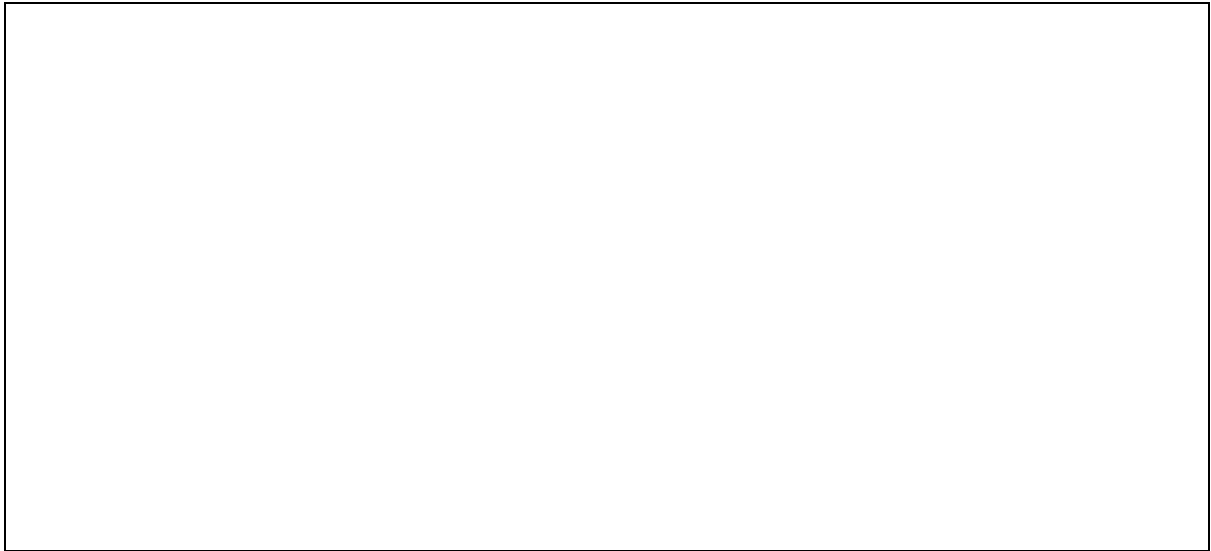
--	--

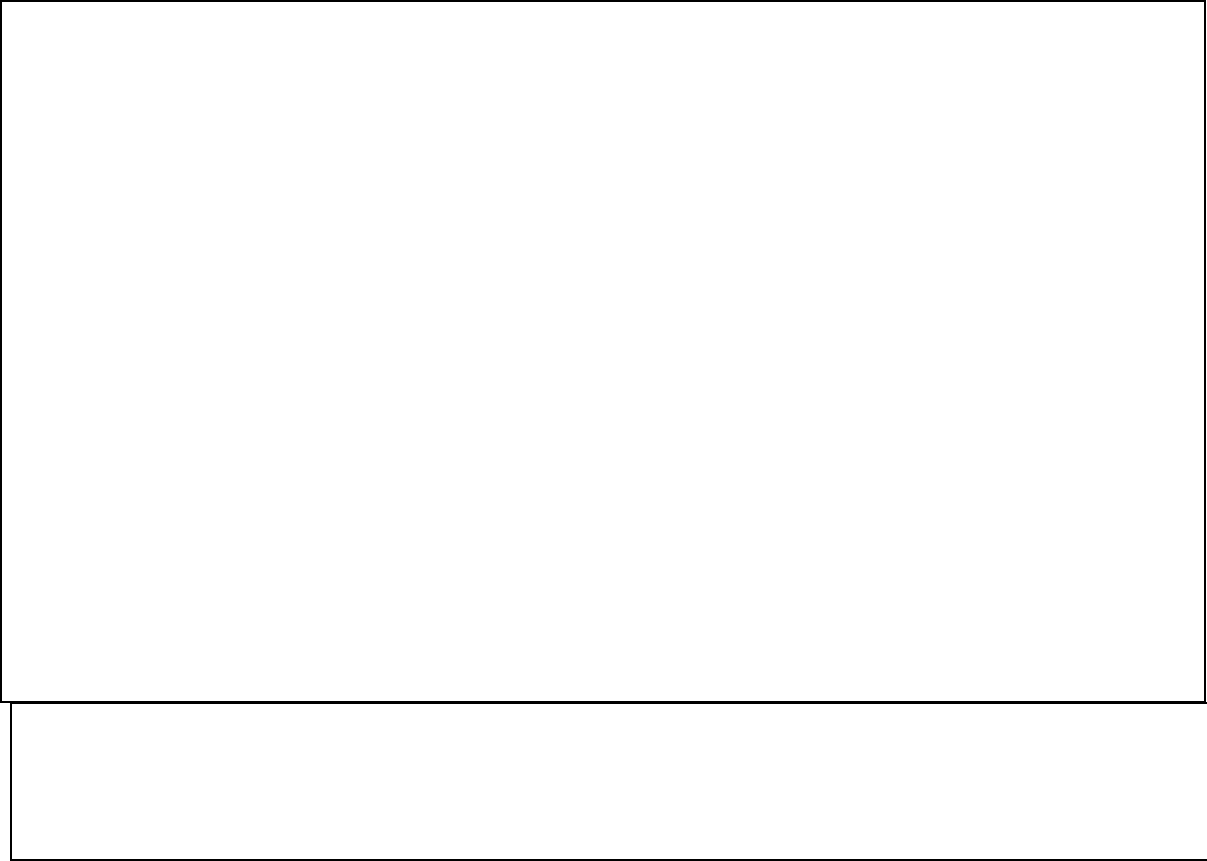
--

--

--	--

--





IV Semester

DESIGN AND ANALYSIS OF ALGORITHMS			
Course Code	21CS42	CIE Marks	50
Teaching Hours/Week (L:T:P: S)	3:0:2:0	SEE Marks	50
Total Hours of Pedagogy	40 T + 20 P	Total Marks	100
Credits	04	Exam Hours	03
Course Learning Objectives:			
<p>CLO 1. Explain the methods of analysing the algorithms and to analyze performance of algorithms. CLO 2. State algorithm's efficiencies using asymptotic notations.</p> <p>CLO 3. Solve problems using algorithm design methods such as the brute force method, greedy method, divide and conquer, decrease and conquer, transform and conquer, dynamic programming, backtracking and branch and bound.</p> <p>CLO 4. Choose the appropriate data structure and algorithm design method for a specified application. CLO 5. Introduce P and NP classes.</p>			
Teaching-Learning Process (General Instructions)			
<p>These are sample Strategies, which teachers can use to accelerate the attainment of the various course outcomes.</p> <ol style="list-style-type: none">1. Lecturer method (L) does not mean only traditional lecture method, but different type of teaching methods may be adopted to develop the outcomes.2. Show Video/animation films to explain functioning of various concepts.3. Encourage collaborative (Group Learning) Learning in the class.4. Ask at least three HOT (Higher order Thinking) questions in the class, which promotes critical thinking.5. Adopt Problem Based Learning (PBL), which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyze information rather than simply recall it.6. Topics will be introduced in a multiple representation.7. Show the different ways to solve the same problem and encourage the students to come up with their own creative ways to solve them.8. Discuss how every concept can be applied to the real world - and when that's possible, it helps improve the students' understanding.			
Module-1			

Introduction: What is an Algorithm? It's Properties. Algorithm Specification-using natural language, using Pseudo code convention, Fundamentals of Algorithmic Problem solving, Analysis Framework Time efficiency and space efficiency, Worst-case, Best-case and Average case efficiency.

Performance Analysis: Estimating Space complexity and Time complexity of algorithms.

Asymptotic Notations: Big-Oh notation (O), Omega notation (Ω), Theta notation (Θ) with examples, Basic efficiency classes, Mathematical analysis of Non-Recursive and Recursive Algorithms with Examples.

Brute force design technique: Selection sort, sequential search, string matching algorithm with complexity Analysis.

Textbook 1: Chapter 1 (Sections 1.1,1.2), Chapter 2(Sections 2.1,2.2,2.3,2.4), Chapter 3(Section 3.1,3.2)

Textbook 2: Chapter 1(section 1.1,1.2,1.3)

Laboratory Component:

- Sort a given set of n integer elements using Selection Sort method and compute its time complexity. Run the program for varied values of $n > 5000$ and record the time taken to sort. Plot a graph of the time taken versus n . The elements can be read from a file or can be generated using the random number generator. Demonstrate using C++/Java how the brute force method works along with its time complexity analysis: worst case, average case and best case.

Teaching-Learning Process

1. Problem based Learning.
2. Chalk & board, Active Learning.
3. Laboratory Demonstration.

Module-2

Divide and Conquer: General method, Recurrence equation for divide and conquer, solving it using Master's theorem. , Divide and Conquer algorithms and complexity Analysis of Finding the maximum & minimum, Binary search, Merge sort, Quick sort.

Decrease and Conquer Approach: Introduction, Insertion sort, Graph searching algorithms, Topological Sorting. It's efficiency analysis.

Textbook 2: Chapter 3(Sections 3.1,3.3,3.4,3.5,3.6)

Textbook 1: Chapter 4 (Sections 4.1,4.2,4.3), Chapter 5(Section 5.1,5.2,5.3)

Laboratory Component:

1. Sort a given set of n integer elements using Quick Sort method and compute its time complexity. Run the program for varied values of n > 5000 and record the time taken to sort. Plot a graph of the time taken versus n. The elements can be read from a file or can be generated using the random number generator. Demonstrate using C++/Java how the divide-and-conquer method works along with its time complexity analysis: worst case, average case and best case.
2. Sort a given set of n integer elements using Merge Sort method and compute its time complexity. Run the program for varied values of n > 5000, and record the time taken to sort. Plot a graph of the time taken versus n. The elements can be read from a file or can be generated using the random number generator. Demonstrate using C++/Java how the divide-and-conquer method works along with its time complexity analysis: worst case, average case and best case.

Teaching-Learning Process

1. Chalk & board, Active Learning, MOOC, Problem based Learning.
2. Laboratory Demonstration.

Module-3

Greedy Method: General method, Coin Change Problem, Knapsack Problem, solving Job sequencing with deadlines Problems.

Minimum cost spanning trees: Prim's Algorithm, Kruskal's Algorithm with performance analysis.

Single source shortest paths: Dijkstra's Algorithm.

Optimal Tree problem: Huffman Trees and Codes.

Transform and Conquer Approach: Introduction, Heaps and Heap Sort.

Textbook 2: Chapter 4(Sections 4.1,4.3,4.5)

Textbook 1: Chapter 9(Section 9.1,9.2,9.3,9.4), Chapter 6(section 6.4)

Laboratory Component:

Write & Execute C++/Java Program

1. To solve Knapsack problem using Greedy method.
2. To find shortest paths to other vertices from a given vertex in a weighted connected graph, using Dijkstra's algorithm.
3. To find Minimum Cost Spanning Tree of a given connected undirected graph using Kruskal's algorithm. Use Union-Find algorithms in your program.
4. To find Minimum Cost Spanning Tree of a given connected undirected graph using Prim's algorithm.

Teaching-Learning Process 1. Chalk

- & board, Active Learning, MOOC, Problem based Learning.
2. Laboratory Demonstration.

Module-4

Dynamic Programming: General method with Examples, Multistage Graphs.

Transitive Closure: Warshall's Algorithm. **All Pairs Shortest Paths:** Floyd's Algorithm, Knapsack problem, Bellman-Ford Algorithm, Travelling Sales Person problem.

Space-Time Tradeoffs: Introduction, Sorting by Counting, Input Enhancement in String Matching, Harspool's algorithm.

Textbook 2: Chapter 5 (Sections 5.1,5.2,5.4,5.9)

Textbook 1: Chapter 8(Sections 8.2,8.4), Chapter 7 (Sections 7.1,7.2)

Laboratory Component:

Write C++/ Java programs to

1. Solve All-Pairs Shortest Paths problem using Floyd's algorithm.
2. Solve Travelling Sales Person problem using Dynamic programming.
3. Solve 0/1 Knapsack problem using Dynamic Programming method.

Teaching-Learning Process

1. Chalk & board, Active Learning, MOOC, Problem based Learning.
2. Laboratory Demonstration.

Module-5

Backtracking: General method, solution using back tracking to N-Queens problem, Sum of subsets problem, Graph coloring, Hamiltonian cycles Problems.

Branch and Bound: Assignment Problem, Travelling Sales Person problem, 0/1 Knapsack problem

NP-Complete and NP-Hard problems: Basic concepts, non- deterministic algorithms, P, NP, NPComplete, and NP-Hard classes.

Textbook 1: Chapter 12 (Sections 12.1,12.2) Chapter 11(11.3)

Textbook 2: Chapter 7 (Sections 7.1,7.2,7.3,7.4,7.5) Chapter 11 (Section 11.1)

Laboratory Component:

1. Design and implement C++/Java Program to find a subset of a given set $S = \{S_1, S_2, \dots, S_n\}$ of n positive integers whose SUM is equal to a given positive integer d . For example, if $S = \{1, 2, 5, 6, 8\}$ and $d = 9$, there are two solutions $\{1, 2, 6\}$ and $\{1, 8\}$. Display a suitable message, if the given problem instance doesn't have a solution.
2. Design and implement C++/Java Program to find all Hamiltonian Cycles in a connected undirected Graph G of n vertices using backtracking principle.

Teaching-Learning Process

1. Chalk & board, Active Learning, MOOC, Problem based learning.
2. Laboratory Demonstration.

Course outcome (Course Skill Set)

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

- CO 1. Analyze the performance of the algorithms, state the efficiency using asymptotic notations and analyze mathematically the complexity of the algorithm.
- CO 2. Apply divide and conquer approaches and decrease and conquer approaches in solving the problems analyze the same
- CO 3. Apply the appropriate algorithmic design technique like greedy method, transform and conquer approaches and compare the efficiency of algorithms to solve the given problem.
- CO 4. Apply and analyze dynamic programming approaches to solve some problems. and improve an algorithm time efficiency by sacrificing space.
- CO 5. Apply and analyze backtracking, branch and bound methods and to describe P, NP and NPComplete problems.

Assessment Details (both CIE and SEE)

The weightage of Continuous Internal Evaluation (CIE) is 50% and for Semester End Exam (SEE) is 50%. The minimum passing mark for the CIE is 40% of the maximum marks (20 marks). A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to each subject/course if the student secures not less than 35% (18 Marks out of 50) in the semester-end examination (SEE), and a minimum of 40% (40 marks out of 100) in the sum total of the CIE (Continuous Internal Evaluation) and SEE (Semester End Examination) taken together

Continuous Internal Evaluation:

Three Unit Tests each of **20 Marks (duration 01 hour)**

1. First test at the end of 5th week of the semester
2. Second test at the end of the 10th week of the semester
3. Third test at the end of the 15th week of the semester

Two assignments each of **10 Marks**

4. First assignment at the end of 4th week of the semester
5. Second assignment at the end of 9th week of the semester

Practical Sessions need to be assessed by appropriate rubrics and viva-voce method. This will contribute to **20 marks**.

- Rubrics for each Experiment taken average for all Lab components – 15 Marks.

- Viva-Voce- 5 Marks (more emphasized on demonstration topics)

The sum of three tests, two assignments, and practical sessions will be out of 100 marks and will be **scaled down to 50 marks**

(to have a less stressed CIE, the portion of the syllabus should not be common /repeated for any of the methods of the CIE. Each method of CIE should have a different syllabus portion of the course).

CIE methods /question paper has to be designed to attain the different levels of Bloom's taxonomy as per the outcome defined for the course.

Semester End Examination:

Theory SEE will be conducted by University as per the scheduled timetable, with common question papers for the subject (**duration 03 hours**)

1. The question paper will have ten questions. Each question is set for 20 marks. Marks scored shall be proportionally reduced to 50 marks
2. There will be 2 questions from each module. Each of the two questions under a module (with a maximum of 3 sub-questions), **should have a mix of topics** under that module.

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

Suggested Learning Resources: Textbooks

1. Introduction to the Design and Analysis of Algorithms, Anany Levitin: 2nd Edition, 2009. Pearson.
2. Computer Algorithms/C++, Ellis Horowitz, SatrajSahni 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)

Weblinks and Video Lectures (e-Resources):

1. <http://elearning.vtu.ac.in/econtent/courses/video/CSE/06CS43.html>
2. <https://nptel.ac.in/courses/106/101/106101060/>
3. <http://elearning.vtu.ac.in/econtent/courses/video/FEP/ADA.html>
4. <http://cse01-iiith.vlabs.ac.in/>
5. <http://openclassroom.stanford.edu/MainFolder/CoursePage.php?course=IntroToAlgorithms>

Activity Based Learning (Suggested Activities in Class)/ Practical Based learning

1. Real world problem solving and puzzles using group discussion. E.g., Fake coin identification, Peasant, wolf, goat, cabbage puzzle, Konigsberg bridge puzzle etc.,
2. Demonstration of solution to a problem through programming.

IV Semester

MICROCONTROLLER AND EMBEDDED SYSTEMS			
Course Code	21CS43	CIE Marks	50
Teaching Hours/Week (L:T:P: S)	3:0:2:0	SEE Marks	50
Total Hours of Pedagogy	40 T + 20 P	Total Marks	100
Credits	04	Exam Hours	03
Course Learning Objectives:			
CLO 1: Understand the fundamentals of ARM-based systems, including programming modules with registers and the CPSR.			
CLO 2: Use the various instructions to program the ARM controller.			
CLO 3: Program various embedded components using the embedded C program.			
CLO 4: Identify various components, their purpose, and their application to the embedded system's applicability.			
CLO 5: Understand the embedded system's real-time operating system and its application in IoT.			
Teaching-Learning Process (General Instructions)			
These are sample Strategies, which teachers can use to accelerate the attainment of the various course outcomes.			
<ol style="list-style-type: none"> 1. The lecturer method (L) does not mean only the traditional lecture method, but different types of teaching methods may be adopted to develop the outcomes. 2. Show video/animation films to explain the functioning of various concepts. 3. Encourage collaborative (group learning) learning in the class. 4. Ask at least three HOT (Higher order Thinking) questions in the class, which promotes critical thinking. 5. Adopt Problem Based Learning (PBL), which fosters students' Analytical skills, develop thinking skills such as the ability to evaluate, generalize, and analyze information rather than simply recall it. 6. Topics will be introduced in multiple representations. 7. Show the different ways to solve the same problem and encourage the students to come up with their own creative ways to solve them. 8. Discuss how every concept can be applied to the real world, and when that's possible, it helps improve the students' understanding. 			
Module-1			
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, Core Extensions			
Textbook 1: Chapter 1 - 1.1 to 1.4, Chapter 2 - 2.1 to 2.5			
Laboratory Component:			
1. Using Keil software, observe the various registers, dump, CPSR, with a simple ALP programme.			
Teaching-Learning Process	<ol style="list-style-type: none"> 1. Demonstration of registers, memory access, and CPSR in a programme module. 2. For concepts, numerical, and discussion, use chalk and a whiteboard, as well as a PowerPoint presentation. 		
Module-2			

Introduction to the ARM Instruction Set: Data Processing Instructions , Branch Instructions, Software Interrupt Instructions, Program Status Register Instructions, Coprocessor Instructions, Loading Constants

C Compilers and Optimization :Basic C Data Types, C Looping Structures, Register Allocation, Function Calls, Pointer Aliasing,

Textbook 1: Chapter 3: Sections 3.1 to 3.6 (Excluding 3.5.2), Chapter 5

Laboratory Component:

2. Write a program to find the sum of the first 10 integer numbers.
3. Write a program to find the factorial of a number.
4. Write a program to add an array of 16 bit numbers and store the 32 bit result in internal RAM.
5. Write a program to find the square of a number (1 to 10) using a look-up table.
6. Write a program to find the largest or smallest number in an array of 32 numbers.

Teaching-Learning Process

1. Demonstration of sample code using Keil software.
2. Laboratory Demonstration

Module-3

C Compilers and Optimization :Structure Arrangement, Bit-fields, Unaligned Data and Endianness, Division, Floating Point, Inline Functions and Inline Assembly, Portability Issues.

ARM programming using Assembly language: Writing Assembly code, Profiling and cycle counting, instruction scheduling, Register Allocation, Conditional Execution, Looping Constructs

Textbook 1: Chapter-5,6

Laboratory Component:

1. Write a program to arrange a series of 32 bit numbers in ascending/descending order.
2. Write a program to count the number of ones and zeros in two consecutive memory locations.
3. Display "Hello World" message using Internal UART.

Teaching-Learning Process

1. Demonstration of sample code using Keil software.
2. Chalk and Board for numerical

Module-4

Embedded System Components: Embedded Vs General computing system, History of embedded systems, Classification of Embedded systems, Major applications areas of embedded systems, purpose of embedded systems.

Core of an Embedded System including all types of processor/controller, Memory, Sensors, Actuators, LED, 7 segment LED display, stepper motor, Keyboard, Push button switch, Communication Interface (onboard and external types), Embedded firmware, Other system components.

Textbook 2: Chapter 1 (Sections 1.2 to 1.6), Chapter 2 (Sections 2.1 to 2.6)

Laboratory Component:

1. Interface and Control a DC Motor.
2. Interface a Stepper motor and rotate it in clockwise and anti-clockwise direction.
3. Determine Digital output for a given Analog input using Internal ADC of ARM controller.
4. Interface a DAC and generate Triangular and Square waveforms.
5. Interface a 4x4 keyboard and display the key code on an LCD.
6. Demonstrate the use of an external interrupt to toggle an LED On/Off.
7. Display the Hex digits 0 to F on a 7-segment LED interface, with an appropriate delay in between.

Teaching-Learning Process

1. Demonstration of sample code for various embedded components using keil.
2. Chalk and Board for numerical and discussion

Module-5

RTOS and IDE for Embedded System Design: Operating System basics, Types of operating systems, Task, process and threads (Only POSIX Threads with an example program), Thread preemption, Multiprocessing and Multitasking, Task Communication (without any program), Task synchronization issues – Racing and Deadlock, Concept of Binary and counting semaphores (Mutex example without any program), How to choose an RTOS, Integration and testing of Embedded hardware and firmware, Embedded system Development Environment – Block diagram (excluding Keil), Disassembler/decompiler, simulator, emulator and debugging techniques, target hardware debugging, boundary scan.

Textbook 2: Chapter-10 (Sections 10.1, 10.2, 10.3, 10.4 , 10.7, 10.8.1.1, 10.8.1.2, 10.8.2.2, 10.10 only), Chapter 12, Chapter-13 (block diagram before 13.1, 13.3, 13.4, 13.5, 13.6 only)

Laboratory Component:

1. Demonstration of IoT applications by using Arduino and Raspberry Pi

Teaching-Learning Process

1. Chalk and Board for numerical and discussion
2. Significance of real time operating system[RTOS] using raspberry pi

Course outcome (Course Skill Set)

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

- CO 1. Explain C-Compilers and optimization
- CO 2. Describe the ARM microcontroller's architectural features and program module.
- CO 3. Apply the knowledge gained from programming on ARM to different applications.
- CO 4. Program the basic hardware components and their application selection method.
- CO 5. Demonstrate the need for a real-time operating system for embedded system applications.

Assessment Details (both CIE and SEE)

The weightage of Continuous Internal Evaluation (CIE) is 50% and for Semester End Exam (SEE) is 50%. The minimum passing mark for the CIE is 40% of the maximum marks (20 marks). A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to each subject/course if the student secures not less than 35% (18 Marks out of 50) in the semester-end examination (SEE), and a minimum of 40% (40 marks out of 100) in the sum total of the CIE (Continuous Internal Evaluation) and SEE (Semester End Examination) taken together **Continuous Internal Evaluation:**

Three Unit Tests each of **20 Marks (duration 01 hour)**

1. First test at the end of 5th week of the semester
2. Second test at the end of the 10th week of the semester
3. Third test at the end of the 15th week of the semester

Two assignments each of **10 Marks**

4. First assignment at the end of 4th week of the semester
5. Second assignment at the end of 9th week of the semester

Practical Sessions need to be assessed by appropriate rubrics and viva-voce method. This will contribute to **20 marks**.

- Rubrics for each Experiment taken average for all Lab components – 15 Marks.
- Viva-Voce– 5 Marks (more emphasized on demonstration topics)

The sum of three tests, two assignments, and practical sessions will be out of 100 marks and will be **scaled down to 50 marks**

(to have a less stressed CIE, the portion of the syllabus should not be common /repeated for any of the methods of the CIE. Each method of CIE should have a different syllabus portion of the course).

CIE methods /question paper has to be designed to attain the different levels of Bloom's taxonomy as per the outcome defined for the course.

Semester End Examination:

Theory SEE will be conducted by University as per the scheduled timetable, with common question papers for the subject (**duration 03 hours**)

1. The question paper will have ten questions. Each question is set for 20 marks. Marks scored shall be proportionally reduced to 50 marks
2. There will be 2 questions from each module. Each of the two questions under a module (with a maximum of 3 sub-questions), **should have a mix of topics** under that module.

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

Suggested Learning Resources:**Textbooks**

1. Andrew N Sloss, Dominic Symes and Chris Wright, ARM system developers guide, Elsevier, Morgan Kaufman publishers, 2008.
2. Shibu K V, "Introduction to Embedded Systems", Tata McGraw Hill Education, Private Limited, 2nd Edition.

Reference Books

1. Raghunandan. G.H, Microcontroller (ARM) and Embedded System, Cengage learning Publication, 2019
2. The Insider's Guide to the ARM7 Based Microcontrollers, Hitex Ltd., 1st edition, 2005.
3. Steve Furber, ARM System-on-Chip Architecture, Second Edition, Pearson, 2015.
4. Raj Kamal, Embedded System, Tata McGraw-Hill Publishers, 2nd Edition, 2008.

Weblinks and Video Lectures (e-Resources):

Activity Based Learning (Suggested Activities in Class)/ Practical Based learning

IV Semester

OPERATING SYSTEMS			
Course Code:	21CS44	CIE Marks	50
Teaching Hours/Week (L:T:P:S)	2:020:0	SEE Marks	50
Total Hours of Pedagogy	40	Total Marks	100
Credits	03	Exam Hours	03
Course Objectives:			
CLO 1. Demonstrate the need for OS and different types of OS CLO 2. Apply suitable techniques for management of different resources CLO 3. Use processor, memory, storage and file system commands CLO 4. Realize the different concepts of OS in platform of usage through case studies			
Teaching-Learning Process (General Instructions)			
These are sample Strategies, which teachers can use to accelerate the attainment of the various course outcomes.			
<ol style="list-style-type: none">1. Lecturer methods (L) need not to be only traditional lecture method, but alternative effective teaching methods could be adopted to attain the outcomes.2. Use of Video/Animation to explain functioning of various concepts.3. Encourage collaborative (Group Learning) Learning in the class.4. Ask at least three HOT (Higher order Thinking) questions in the class, which promotes critical thinking.5. Adopt Problem Based Learning (PBL), which fosters students' Analytical skills, develop design thinking skills such as the ability to design, evaluate, generalize, and analyze information rather than simply recall it.6. Introduce Topics in manifold representations.7. Show the different ways to solve the same problem and encourage the students to come up with their own creative ways to solve them.8. Discuss how every concept can be applied to the real world - and when that's possible, it helps improve the students' understanding.			
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

Textbook 1: Chapter - 1,2,3

Teaching-Learning Process	<p>Active learning and problem solving</p> <ol style="list-style-type: none"> 1. https://www.youtube.com/watch?v=vBURTt97EkA&list=PLBlNk6fEYqRiVhbXDGLXdk_OQAeuVcp20 2. https://www.youtube.com/watch?v=a2B69vCtjOU&list=PL3wYxht4yCjpcfUDz-TgD_ainZ2K3MUZ&index=2
----------------------------------	---

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.

Process Synchronization: Synchronization: The critical section problem; Peterson's solution; Synchronization hardware; Semaphores; Classical problems of synchronization; Monitors.

Textbook 1: Chapter - 4,5

Teaching-Learning Process	<p>Active Learning and problem solving</p> <ol style="list-style-type: none"> 1. https://www.youtube.com/watch?v=HW2Wcx-ktsc 2. https://www.youtube.com/watch?v=9YRxhlvt9Zo
----------------------------------	---

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.

Textbook 1: Chapter - 7,8

Teaching-Learning Process	<p>Active Learning, Problem solving based on deadlock with animation</p> <ol style="list-style-type: none"> 1. https://www.youtube.com/watch?v=MYgmmJfdBg 2. https://www.youtube.com/watch?v=Y14b7_T3AEw&list=PLEJxKK7AcSEGPOCftQTJhOElU44JJAun&index=30
----------------------------------	---

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.

Textbook 1: Chapter - 9,10,11

Teaching-Learning Process

Active learning about memory management and File system

1. <https://www.youtube.com/watch?v=pJ6qrCB8pDw&list=PLIY8eNdw5tW-BxRY0yK3fYTYVqytw8qhp>
2. <https://www.youtube.com/watch?v=-orfFhvNBzY>

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.

Textbook 1: Chapter - 2,21

Teaching-Learning Process

Active learning about case studies

1. <https://www.youtube.com/watch?v=TTBkc5eiju4>
2. https://www.youtube.com/watch?v=8hkvMRGTzCM&list=PLEAYkSg4uSQ2PAch478muxnoeTNz_QeUJ&index=36
3. <https://www.youtube.com/watch?v=mX1FEur4VCw>

Course Outcomes (Course Skill Set)

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

- CO 1. Identify the structure of an operating system and its scheduling mechanism.
- CO 2. Demonstrate the allocation of resources for a process using scheduling algorithm.
- CO 3. Identify root causes of deadlock and provide the solution for deadlock elimination.
- CO 4. Explore about the storage structures and learn about the Linux Operating system.
- CO 5. Analyze Storage Structures and Implement Customized Case study

Assessment Details (both CIE and SEE)

The weightage of Continuous Internal Evaluation (CIE) is 50% and for Semester End Exam (SEE) is 50%. The minimum passing mark for the CIE is 40% of the maximum marks (20 marks). A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to each subject/course if the student secures not less than 35% (18 Marks out of 50) in the semester-end examination (SEE), and a minimum of 40% (40 marks out of 100) in the sum total of the CIE (Continuous Internal Evaluation) and SEE (Semester End Examination) taken together **Continuous Internal Evaluation:**

Three Unit Tests each of **20 Marks (duration 01 hour)**

1. First test at the end of 5th week of the semester
2. Second test at the end of the 10th week of the semester
3. Third test at the end of the 15th week of the semester

Two assignments each of **10 Marks**

4. First assignment at the end of 4th week of the semester
5. Second assignment at the end of 9th week of the semester

Group discussion/Seminar/quiz any one of three suitably planned to attain the COs and POs for **20 Marks (duration 01 hours)**

6. At the end of the 13th week of the semester

The sum of three tests, two assignments, and quiz/seminar/group discussion will be out of 100 marks and will be **scaled down to 50 marks**

(to have less stressed CIE, the portion of the syllabus should not be common /repeated for any of the methods of the CIE. Each method of CIE should have a different syllabus portion of the course).

CIE methods /question paper has to be designed to attain the different levels of Bloom's taxonomy as per the outcome defined for the course.

Semester End Examination:

Theory SEE will be conducted by University as per the scheduled timetable, with common question papers for the subject **(duration 03 hours)**

1. The question paper will have ten questions. Each question is set for 20 marks. Marks scored shall be proportionally reduced to 50 marks
2. There will be 2 questions from each module. Each of the two questions under a module (with a maximum of 3 sub-questions), **should have a mix of topics** under that module.

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

Suggested Learning Resources:**Textbooks**

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.

Weblinks and Video Lectures (e-Resources):

1. https://www.youtube.com/watch?v=vBURTt97EkA&list=PLBlnK6fEyyqRiVhbXDGLXDk_OQAe_uVcp20
2. https://www.youtube.com/watch?v=783KAB-tuE4&list=PLlemF3uozcAKTgsClj82voMK3TMR0YE_f
3. <https://www.youtube.com/watch?v=3-ITLMMeeXY&list=PL3pGy4HtqwD0n7bQfHjPnsWzkeR-n6mkO>

Activity Based Learning (Suggested Activities in Class)/ Practical Based learning □

Real world problem solving using group discussion.

- Role play for process scheduling.
- Present animation for Deadlock.
- Real world examples of memory management concepts

IV Semester

PYTHON PROGRAMMING LABORATORY			
Course Code	21CSL46	CIE Marks	50
Teaching Hours/Weeks (L: T: P: S)	0: 0: 2: 0	SEE Marks	50
Total Hours of Pedagogy	24	Total Marks	100
Credits	01	Exam Hours	03
Course Objectives:			
CLO 1. Demonstrate the use of IDLE or PyCharm IDE to create Python Applications			
CLO 2. Using Python programming language to develop programs for solving real-world problems			
CLO 3. Implement the Object-Oriented Programming concepts in Python.			
CLO 4. Appraise the need for working with various documents like Excel, PDF, Word and Others			
CLO 5. Demonstrate regular expression using python programming			
Note: two hours tutorial is suggested for each laboratory sessions.			
Prerequisite			
<ul style="list-style-type: none"> Students should be familiarized about Python installation and setting Python environment Usage of IDLE or IDE like PyCharm should be introduced Python Installation: https://www.youtube.com/watch?v=Kn1HF3oD19c PyCharm Installation: https://www.youtube.com/watch?v=SZUNUB6nz3g 			
Sl. No.	<i>PART A - List of problems for which student should develop program and execute in the Laboratory</i>		
1	<p>Aim: Introduce the Python fundamentals, data types, operators, flow control and exception handling in Python</p> <p>a) Write a python program to find the best of two test average marks out of three test's marks accepted from the user.</p> <p>b) Develop a Python program to check whether a given number is palindrome or not and also count the number of occurrences of each digit in the input number.</p> <p>Datatypes: https://www.youtube.com/watch?v=gCCVsvgR2KU Operators: https://www.youtube.com/watch?v=v5MR5JnKcZI Flow Control: https://www.youtube.com/watch?v=PqFKRqpHrjw For loop: https://www.youtube.com/watch?v=0ZvaDa8eT5s While loop: https://www.youtube.com/watch?v=HZARImviDxg Exceptions: https://www.youtube.com/watch?v=6SPDvPK38tw</p>		
2	<p>Aim: Demonstrating creation of functions, passing parameters and return values</p> <p>a) Defined as a function F as $F_n = F_{n-1} + F_{n-2}$. Write a Python program which accepts a value for N (where $N > 0$) as input and pass this value to the function. Display suitable error message if the condition for input value is not followed.</p> <p>b) Develop a python program to convert binary to decimal, octal to hexadecimal using functions.</p> <p>Functions: https://www.youtube.com/watch?v=BVfcWuca9nw Arguments: https://www.youtube.com/watch?v=ijXMGpoMkhQ Return value: https://www.youtube.com/watch?v=nuNXiEDnM44</p>		

6	<p>Aim: Demonstration of reading, writing and organizing files.</p> <p>a) Write a python program to accept a file name from the user and perform the following operations</p> <ol style="list-style-type: none"> 1. Display the first N line of the file 2. Find the frequency of occurrence of the word accepted from the user in the file <p>b) Write a python program to create a ZIP file of a particular folder which contains several files inside it.</p> <p>Files: https://www.youtube.com/watch?v=vuyb7CxZgbU https://www.youtube.com/watch?v=FqcjKewJTQ0</p> <p>File organization: https://www.youtube.com/watch?v=MRuq3SRXses</p>
7	<p>Aim: Demonstration of the concepts of classes, methods, objects and inheritance</p>

	<p>a) By using the concept of inheritance write a python program to find the area of triangle, circle and rectangle.</p> <p>b) Write a python program by creating a class called Employee to store the details of Name, Employee_ID, Department and Salary, and implement a method to update salary of employees belonging to a given department.</p> <p>OOP's concepts: https://www.youtube.com/watch?v=qiSCMNBIP2g Inheritance: https://www.youtube.com/watch?v=Cn7AkDb4pIU</p>
8	<p>Aim: Demonstration of classes and methods with polymorphism and overriding</p> <p>a) Write a python program to find the whether the given input is palindrome or not (for both string and integer) using the concept of polymorphism and inheritance.</p> <p>Overriding: https://www.youtube.com/watch?v=CcTzTuIsoFk</p>
9	<p>Aim: Demonstration of working with excel spreadsheets and web scraping</p> <p>a) Write a python program to download the all XKCD comics</p> <p>b) Demonstrate python program to read the data from the spreadsheet and write the data in to the spreadsheet</p> <p>Web scraping: https://www.youtube.com/watch?v=ng2o98k983k</p> <p>Excel: https://www.youtube.com/watch?v=nsKNPHJ9iPc</p>

10	<p>Aim: Demonstration of working with PDF, word and JSON files</p> <p>a) Write a python program to combine select pages from many PDFs b) Write a python program to fetch current weather data from the JSON file</p> <p>PDFs: https://www.youtube.com/watch?v=q70xzDG6nls https://www.youtube.com/watch?v=JhQVD7Y1bsA https://www.youtube.com/watch?v=FcrW-ESdY-A</p> <p>Word files: https://www.youtube.com/watch?v=ZU3cSl51jWE</p> <p>JSON files: https://www.youtube.com/watch?v=9N6a-VLBa2I</p>
<p>Python (Full Course): https://www.youtube.com/watch?v=_uQrJ0TkZlc</p>	
<p>Pedagogy</p>	<p>For the above experiments the following pedagogy can be considered. Problem based learning, Active learning, MOOC, Chalk &Talk</p>
<p style="text-align: center;">PART B – Practical Based Learning</p>	
<p>A problem statement for each batch is to be generated in consultation with the co-examiner and student should develop an algorithm, program and execute the program for the given problem with appropriate outputs.</p>	
<p>Course Outcomes:</p> <p>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. CO 5. Determine the need for scraping websites and working with PDF, JSON and other file formats.</p>	

Assessment Details (both CIE and SEE)

The weightage of Continuous Internal Evaluation (CIE) is 50% and for Semester End Exam (SEE) is 50%. The minimum passing mark for the CIE is 40% of the maximum marks (20 marks). A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to each course. The student has to secure not less than 35% (18 Marks out of 50) in the semester-end examination (SEE). The student has to secure 40% of sum of the maximum marks of CIE and SEE to qualify in the course.

Continuous Internal Evaluation (CIE):

CIE marks for the practical course is **50 Marks**.

The split-up of CIE marks for record/ journal and test are in the ratio **60:40**.

- Each experiment to be evaluated for conduction with observation sheet and record write-up. Rubrics for the evaluation of the journal/write-up for hardware/software experiments designed by the faculty who is handling the laboratory session and is made known to students at the beginning of the practical session.
- Record should contain all the specified experiments in the syllabus and each experiment write-up will be evaluated for 10 marks.
- Total marks scored by the students are scaled down to 30 marks (60% of maximum marks).
- Weightage to be given for neatness and submission of record/write-up on time.
- Department shall conduct 02 tests for 100 marks, the first test shall be conducted after the 8th week of the semester and the second test shall be conducted after the 14th week of the semester.
- In each test, test write-up, conduction of experiment, acceptable result, and procedural knowledge will carry a weightage of 60% and the rest 40% for viva-voce.
- The suitable rubrics can be designed to evaluate each student's performance and learning ability. Rubrics suggested in Annexure-II of Regulation book
- The average of 02 tests is scaled down to **20 marks** (40% of the maximum marks).

The Sum of scaled-down marks scored in the report write-up/journal and average marks of two tests is the total CIE marks scored by the student.

Semester End Evaluation (SEE):

- SEE marks for the practical course is 50 Marks.
- SEE shall be conducted jointly by the two examiners of the same institute, examiners are appointed by the University
- All laboratory experiments are to be included for practical examination.
- (Rubrics) Breakup of marks and the instructions printed on the cover page of the answer script to be strictly adhered to by the examiners. **OR** based on the course requirement evaluation rubrics shall be decided jointly by examiners.
- Students can pick one question (experiment) from the questions lot prepared by the internal /external examiners jointly.
- Evaluation of test write-up/ conduction procedure and result/viva will be conducted jointly by examiners.
- General rubrics suggested for SEE are mentioned here, writeup-20%, Conduction procedure and result in -60%, Viva-voce 20% of maximum marks. SEE for practical shall be evaluated for 100 marks and scored marks shall be scaled down to 50 marks (however, based on course type, rubrics shall be decided by the examiners)
- *Students can pick one experiment from the questions lot of PART A with equal choice to all the students in a batch. For PART B examiners should frame a question for each batch, student should*

develop an algorithm, program, execute and demonstrate the results with appropriate output for the given problem.

- *Weightage of marks for PART A is 80% and for PART B is 20%. General rubrics suggested to be followed for part A and part B.*
- Change of experiment is allowed only once and Marks allotted to the procedure part to be made zero (Not allowed for Part B).
- The duration of SEE is 03 hours

Rubrics suggested in Annexure-II of Regulation book

Textbooks:

1. Al Sweigart, **“Automate the Boring Stuff with Python”**,1stEdition, No Starch Press, 2015. (Available under CC-BY-NC-SA license at <https://automatetheboringstuff.com/>)
2. Reema Thareja **“Python Programming Using Problem Solving Approach”** Oxford University Press.
3. Allen B. Downey, **“Think Python: How to Think Like a Computer Scientist”**, 2nd Edition, Green Tea Press, 2015. (Available under CC-BY-NC license at <http://greenteapress.com/thinkpython2/thinkpython2.pdf>)

IV Semester

WEB PROGRAMMING (Practical based)			
Course Code	21CSL481	CIE Marks	50
Teaching Hours/Week (L:T:P: S)	1:0:0:0	SEE Marks	50
Total Hours of Pedagogy	12T + 12P	Total Marks	100
Credits	01	Exam Hours	02
Course Objectives:			
<p>CLO 1. Learn Web tool box and history of web browsers.</p> <p>CLO 2. Learn HTML, XHTML tags with utilizations.</p> <p>CLO 3. Know CSS with dynamic document utilizations.</p> <p>CLO 4. Learn JavaScript with Element access in JavaScript.</p> <p>CLO 5. Logically plan and develop web pages..</p>			
Teaching-Learning Process (General Instructions)			
<p>These are sample Strategies, which teachers can use to accelerate the attainment of the various course outcomes.</p> <ol style="list-style-type: none"> 1. Lecturer method (L) need not to be only a traditional lecture method, but alternative effective teaching methods could be adopted to attain the outcomes. 2. Use of Video/Animation to explain functioning of various concepts. 3. Encourage collaborative (Group Learning) Learning in the class. 4. Ask at least three HOT (Higher order Thinking) questions in the class, which promotes critical thinking. 5. Adopt Problem Based Learning (PBL), which fosters students' Analytical skills, develop design thinking skills such as the ability to design, evaluate, generalize, and analyze information rather than simply recall it. 6. Introduce Topics in manifold representations. 7. Show the different ways to solve the same problem with different circuits/logic and encourage the students to come up with their own creative ways to solve them. 8. Discuss how every concept can be applied to the real world - and when that's possible, it helps improve the students' understanding. 			
Module-1			
Introduction to WEB Programming: Internet, WWW, Web Browsers, and Web Servers, URLs, MIME, HTTP, Security, The Web Programmers Toolbox.			
Textbook 1: Chapter 1(1.1 to 1.9)			
Teaching-Learning Process	Chalk and board, Active Learning, practical based learning		
Module-2			
HTML and XHTML: Origins of HTML and XHTML, Basic syntax, Standard XHTML document structure, Basic text markup, Images, Hypertext Links, Lists, Tables. Forms, Frames in HTML and XHTML, Syntactic differences between HTML and XHTML.			
Textbook 1: Chapter 2(2.1 to 2.10)			
Teaching-Learning Process	Chalk and board, Active Learning, Demonstration, presentation, problem solving		
Module-3			

CSS: Introduction, Levels of style sheets, Style specification formats, Selector forms, Property value forms, Font properties, List properties, Color, Alignment of text, Background images, tags.

Textbook 1: Chapter 3(3.1 to 3.12)

Teaching-Learning Process	Chalk and board, Demonstration, problem solving
----------------------------------	---

Module-4

Java Script - I: Object orientation and JavaScript; General syntactic characteristics; Primitives, Operations, and expressions; Screen output and keyboard input.

Textbook 1: Chapter 4(4.1 to 4.5)

Teaching-Learning Process	Chalk and board, Practical based learning, practical's
----------------------------------	--

Module-5

Java Script - II: Control statements, Object creation and Modification; Arrays; Functions; Constructor; Pattern matching using expressions; Errors, Element access in JavaScript.

Textbook 1: Chapter 4(4.6 to 4.14)

Teaching-Learning Process	Chalk and board, MOOC
----------------------------------	-----------------------

Course Outcomes (Course Skill Set):

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

- CO 1. Describe the fundamentals of web and concept of HTML.
- CO 2. Use the concepts of HTML, XHTML to construct the web pages.
- CO 3. Interpret CSS for dynamic documents.
- CO 4. Evaluate different concepts of JavaScript & Construct dynamic documents.
- CO 5. Design a small project with JavaScript and XHTML.

Assessment Details (both CIE and SEE)

The weightage of Continuous Internal Evaluation (CIE) is 50% and for Semester End Exam (SEE) is 50%. The minimum passing mark for the CIE is 40% of the maximum marks (20 marks). A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to each course. The student has to secure not less than 35% (18 Marks out of 50) in the semester-end examination (SEE).

Continuous Internal Evaluation (CIE):

NOTE: List of experiments to be prepared by the faculty based on the syllabus mentioned above CIE marks for the practical course is **50 Marks**.

The split-up of CIE marks for record/ journal and test are in the ratio **60:40**.

- Each experiment to be evaluated for conduction with observation sheet and record write-up. Rubrics for the evaluation of the journal/write-up for hardware/software experiments designed by the faculty who is handling the laboratory session and is made known to students at the beginning of the practical session.
- Record should contain all the specified experiments in the syllabus and each experiment write-up will be evaluated for 10 marks.
- Total marks scored by the students are scaled down to 30 marks (60% of maximum marks).
- Weightage to be given for neatness and submission of record/write-up on time.
- Department shall conduct 02 tests for 100 marks, the first test shall be conducted after the 8th week of the semester and the second test shall be conducted after the 14th week of the semester.
- In each test, test write-up, conduction of experiment, acceptable result, and procedural knowledge will carry a weightage of 60% and the rest 40% for viva-voce.
- The suitable rubrics can be designed to evaluate each student's performance and learning ability. Rubrics suggested in Annexure-II of Regulation book
- The average of 02 tests is scaled down to **20 marks** (40% of the maximum marks).

The Sum of scaled-down marks scored in the report write-up/journal and average marks of two tests is the total CIE marks scored by the student.

Semester End Evaluation (SEE):

- SEE marks for the practical course is 50 Marks.
- SEE shall be conducted jointly by the two examiners of the same institute, examiners are appointed by the University
- All laboratory experiments are to be included for practical examination.

- (Rubrics) Breakup of marks and the instructions printed on the cover page of the answer script to be strictly adhered to by the examiners. **OR** based on the course requirement evaluation rubrics shall be decided jointly by examiners.
- Students can pick one question (experiment) from the questions lot prepared by the internal /external examiners jointly.
- Evaluation of test write-up/ conduction procedure and result/viva will be conducted jointly by examiners.
- General rubrics suggested for SEE are mentioned here, writeup-20%, Conduction procedure and result in -60%, Viva-voce 20% of maximum marks. SEE for practical shall be evaluated for 100 marks and scored marks shall be scaled down to 50 marks (however, based on course type, rubrics shall be decided by the examiners)
- The duration of SEE is 02 hours

Rubrics suggested in Annexure-II of Regulation book

Textbooks

1. Robert W Sebesta, "Programming the World Wide Web", 6th Edition, Pearson Education, 2008.

Reference Books

1. M.Deitel, P.J.Deitel, A.B.Goldberg, "Internet & World Wide Web How to program", 3rd Edition, Pearson Education / PHI, 2004.
2. Chris Bates, "Web Programming Building Internet Applications", 3rd Edition, Wiley India, 2006.
3. Xue Bai et al, "The Web Warrior Guide to Web Programming", Thomson, 2003.
4. Sklar, "The Web Warrior Guide to Web Design Technologies", 1st Edition, Cengage Learning India

Weblinks and Video Lectures (e-Resources):

1. Fundamentals of WEB Programming: <https://www.youtube.com/watch?v=DR9dr6gxdM>
2. HTML and XHTML: <https://www.youtube.com/watch?v=A1XIIDDxgwg>
3. CSS: <https://www.youtube.com/watch?v=J35jug1uHzE>
4. Java Script and HTML Documents: <https://www.youtube.com/watch?v=Gd0RBdFRvF0>
5. Dynamic Documents with JavaScript: <https://www.youtube.com/watch?v=HTFSIJALNKc>

Tutorial Link:

1. <http://www.tutorialspoint.com>
2. <http://www.w3schools.com>

Activity Based Learning (Suggested Activities in Class)/ Practical Based learning

- Demonstration of simple projects

IV Semester

UNIX SHELL PROGRAMMING			
Course Code	21CS482	CIE Marks	50
Teaching Hours/Week (L:T:P: S)	1:0:0:0	SEE Marks	50
Total Hours of Pedagogy	12	Total Marks	100
Credits	01	Exam Hours	01
Course Objectives:			
CLO 1. To help the students to understand effective use of Unix concepts, commands and terminology.			
CLO 2. Identify, access, and evaluate UNIX file system.			
CLO 3. Understand UNIX command syntax and semantics.			
CLO 4. Ability to read and understand specifications, scripts and programs.			
CLO 5. Analyze Facility with UNIX Process.			

Teaching-Learning Process (General Instructions)

These are sample Strategies, which teachers can use to accelerate the attainment of the various course outcomes.

1. Lecturer method (L) need not to be only a traditional lecture method, but alternative effective teaching methods could be adopted to attain the outcomes.
2. Use of Video/Animation to explain functioning of various concepts.
3. Encourage collaborative (Group Learning) Learning in the class.
4. Ask at least three HOT (Higher order Thinking) questions in the class, which promotes critical thinking.
5. Adopt Problem Based Learning (PBL), which fosters students' Analytical skills, develop design thinking skills such as the ability to design, evaluate, generalize, and analyze information rather than simply recall it.
6. Introduce Topics in manifold representations.
7. Show the different ways to solve the same problem with different circuits/logic and encourage the students to come up with their own creative ways to solve them.
8. Discuss how every concept can be applied to the real world - and when that's possible, it helps improve the students' understanding.

Module-1

Introduction of UNIX - Introduction, History, Architecture, Experience the Unix environment, Basic commands ls, cat, cal, date, calendar, who, printf, tty, sty, uname, passwd, echo, tput, and bc.

Textbook 1: Chapter 1(1.1 to 1.4) , Chapter 2- 2.1

Teaching-Learning Process	Chalk and board, Active Learning, practical based learning
----------------------------------	--

Module-2

UNIX File System- The file, what's in a filename? The parent-child relationship, pwd, the Home directory, absolute pathnames, using absolute pathnames for a command, cd, mkdir, rmdir, Relative pathnames, The UNIX file system.

Textbook 1: Chapter 4

Teaching-Learning Process	Chalk and board, Active Learning, Demonstration, presentation, problem solving
----------------------------------	--

Module-3

Basic File Attributes - ls - l, the -d option, File Permissions, chmod, Security and File Permission, users and groups, security level, changing permission, user masks, changing ownership and group, File Attributes, More file attributes: hard link, symbolic link, umask, find.

Textbook 1: Chapter 6

Teaching-Learning Process	Chalk and board, Demonstration, problem solving
----------------------------------	---

Module-4

Introduction to the Shell Scripting - Introduction to Shell Scripting, Shell Scripts, read, Command Line Arguments, Exit Status of a Command, The Logical Operators && and ||, exit, if, and case conditions, expr, sleep and wait, while, until, for, \$, @, redirection. The here document, set, trap, Sample Validation and Data Entry Scripts.

Textbook 1: Chapter 11,12,14

Teaching-Learning Process	Chalk and board, Practical based learning, practical's
----------------------------------	--

Module-5

Introduction to UNIX System process: Mechanism of process creation. Parent and child process. The ps command with its options. Executing a command at a specified point of time: at command. Executing a command periodically: cron command and the crontab file.. Signals.

Textbook 1: Chapter 9,19

Teaching-Learning Process	Chalk and board, MOOC
----------------------------------	-----------------------

Course Outcomes (Course Skill Set):

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

- CO 1. Know the basics of Unix concepts and commands.
- CO 2. Evaluate the UNIX file system.
- CO 3. Apply Changes in file system.
- CO 4. Understand scripts and programs.
- CO 5. Analyze Facility with UNIX system process

Assessment Details (both CIE and SEE)

The weightage of Continuous Internal Evaluation (CIE) is 50% and for Semester End Exam (SEE) is 50%. The minimum passing mark for the CIE is 40% of the maximum marks (20 marks). A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to each subject/course if the student secures not less than 35% (18 Marks out of 50) in the semester-end examination (SEE), and a minimum of 40% (40 marks out of 100) in the sum total of the CIE (Continuous Internal Evaluation) and SEE (Semester End Examination) taken together.

Continuous Internal Evaluation:

Three Unit Tests each of **20 Marks (duration 01 hour)**

- 1. First test at the end of 5th week of the semester
- 2. Second test at the end of the 10th week of the semester
- 3. Third test at the end of the 15th week of the semester

Two assignments each of **10 Marks**

- 4. First assignment at the end of 4th week of the semester
- 5. Second assignment at the end of 9th week of the semester

Group discussion/Seminar/quiz any one of three suitably planned to attain the COs and POs for **20 Marks (duration 01 hours)**

- 6. At the end of the 13th week of the semester

The sum of three tests, two assignments, and quiz/seminar/group discussion will be out of 100 marks and will be **scaled down to 50 marks**

(to have less stressed CIE, the portion of the syllabus should not be common /repeated for any of the methods of the CIE. Each method of CIE should have a different syllabus portion of the course).

CIE methods /question paper has to be designed to attain the different levels of Bloom's taxonomy as per the outcome defined for the course.

Semester End Examination:

Theory SEE will be conducted by University as per the scheduled timetable, with common question papers for the subject (**duration 01 hours**)

SEE paper will be set for 50 questions of each of 01 marks. The pattern of the question paper is MCQ. The time allotted for SEE is 01 hours

Textbooks

- 1. Unix Concepts & Applications 4th Edition, Sumitabha Das, Tata McGraw Hill References:
- 2. Unix Shell Programming, Yashwant Kanetkar 3. Introduction to UNIX by M G Venkatesh Murthy.

Weblinks and Video Lectures (e-Resources):

1. <https://www.youtube.com/watch?v=ffYUfAqEamY>
2. <https://www.youtube.com/watch?v=Q05NZiYFcD0>
3. <https://www.youtube.com/watch?v=8GdT53KDIyY>
4. <https://www.youtube.com/watch?app=desktop&v=3Pga3y7rCgo>

Activity Based Learning (Suggested Activities in Class)/ Practical Based learning □

Real world problem solving using group discussion.

- Real world examples of Linux operating system Utilizations.

IV Semester

R PROGRAMMING (Practical based)			
Course Code	21CSL483	CIE Marks	50
Teaching Hours/Week (L:T:P: S)	1:0:0:0	SEE Marks	50
Total Hours of Pedagogy	12T + 12P	Total Marks	100
Credits	01	Exam Hours	02
Course Objectives:			
<p>CLO 1. Explore and understand how R and R Studio interactive environment.</p> <p>CLO 2. To learn and practice programming techniques using R programming.</p> <p>CLO 3. Read Structured Data into R from various sources.</p> <p>CLO 4. Understand the different data Structures, data types in R. CLO</p> <p>5. To develop small applications using R Programming</p>			
Teaching-Learning Process (General Instructions)			
<p>These are sample Strategies, which teachers can use to accelerate the attainment of the various course outcomes.</p> <ol style="list-style-type: none"> 1. Lecturer method (L) need not to be only a traditional lecture method, but alternative effective teaching methods could be adopted to attain the outcomes. 2. Use of Video/Animation to explain functioning of various concepts. 3. Encourage collaborative (Group Learning) Learning in the class. 4. Ask at least three HOT (Higher order Thinking) questions in the class, which promotes critical thinking. 5. Adopt Problem Based Learning (PBL), which fosters students' Analytical skills, develop design thinking skills such as the ability to design, evaluate, generalize, and analyze information rather than simply recall it. 6. Introduce Topics in manifold representations. 7. Show the different ways to solve the same problem with different circuits/logic and encourage the students to come up with their own creative ways to solve them. 8. Discuss how every concept can be applied to the real world - and when that's possible, it helps improve the students' understanding. 			
Module-1			
Numeric, Arithmetic, Assignment, and Vectors: R for Basic Math, Arithmetic, Variables, Functions, Vectors, Expressions and assignments Logical expressions.			
Textbook 1: Chapter 2(2.1 to 2.7)			
Teaching-Learning Process	Chalk and board, Active Learning, practical based learning		
Module-2			
Matrices and Arrays: Defining a Matrix, Sub-setting, Matrix Operations, Conditions and Looping: if statements, looping with for, looping with while, vector based programming.			
Textbook 1: Chapter 2- 2.8, chapter 3- 3.2 to 3.5			
Teaching-Learning Process	Chalk and board, Active Learning, Demonstration, presentation, problem solving		
Module-3			
Lists and Data Frames: Data Frames, Lists , Special values, The apply family.			
Textbook 1: Chapter 6- 6.2 to 6.4			
Teaching-Learning Process	Chalk and board, Demonstration, problem solving		
Module-4			

Functions: Calling functions, scoping, Arguments matching, writing functions: The function command, Arguments, specialized function.

Textbook 1: Chapter 5- 5.1 to 5.6

Teaching-Learning Process

Chalk and board, Practical based learning, practical's

Module-5

Pointers: packages, frames, de bugging, manipulation of code, compilation of the code.

Textbook 1: Chapter 8- 8.1 to 8.8

Teaching-Learning Process

Chalk and board, MOOC

Course Outcomes (Course Skill Set):

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

- CO 1. To understand the fundamental syntax of R through readings, practice exercises, CO
- 2. To demonstrations, and writing R code.
- CO 3. To apply critical programming language concepts such as data types, iteration,
- CO 4. To understand control structures, functions, and Boolean operators by writing R programs and through examples
- CO 5. To import a variety of data formats into R using R-Studio CO
- 6. To prepare or tidy data for in preparation for analyze.

Assessment Details (both CIE and SEE)

The weightage of Continuous Internal Evaluation (CIE) is 50% and for Semester End Exam (SEE) is 50%. The minimum passing mark for the CIE is 40% of the maximum marks (20 marks). A student shall be deemed to have satisfied the academic requirements and earned the credits allotted to each course. The student has to secure not less than 35% (18 Marks out of 50) in the semester-end examination (SEE).

Continuous Internal Evaluation (CIE):

NOTE: List of experiments to be prepared by the faculty based on the syllabus mentioned above CIE marks for the practical course is **50 Marks**.

The split-up of CIE marks for record/ journal and test are in the ratio **60:40**.

- Each experiment to be evaluated for conduction with observation sheet and record write-up. Rubrics for the evaluation of the journal/write-up for hardware/software experiments designed by the faculty who is handling the laboratory session and is made known to students at the beginning of the practical session.
- Record should contain all the specified experiments in the syllabus and each experiment write-up will be evaluated for 10 marks.
- Total marks scored by the students are scaled down to 30 marks (60% of maximum marks).
- Weightage to be given for neatness and submission of record/write-up on time.
- Department shall conduct 02 tests for 100 marks, the first test shall be conducted after the 8th week of the semester and the second test shall be conducted after the 14th week of the semester.
- In each test, test write-up, conduction of experiment, acceptable result, and procedural knowledge will carry a weightage of 60% and the rest 40% for viva-voce.
- The suitable rubrics can be designed to evaluate each student's performance and learning ability. Rubrics suggested in Annexure-II of Regulation book
- The average of 02 tests is scaled down to **20 marks** (40% of the maximum marks).

The Sum of scaled-down marks scored in the report write-up/journal and average marks of two tests is the total CIE marks scored by the student.

Semester End Evaluation (SEE):

- SEE marks for the practical course is 50 Marks.
- SEE shall be conducted jointly by the two examiners of the same institute, examiners are appointed by the University
- All laboratory experiments are to be included for practical examination.
- (Rubrics) Breakup of marks and the instructions printed on the cover page of the answer script to be strictly adhered to by the examiners. **OR** based on the course requirement evaluation rubrics shall be decided jointly by examiners.

- Students can pick one question (experiment) from the questions lot prepared by the internal /external examiners jointly.
- Evaluation of test write-up/ conduction procedure and result/viva will be conducted jointly by examiners.
- General rubrics suggested for SEE are mentioned here, writeup-20%, Conduction procedure and result in -60%, Viva-voce 20% of maximum marks. SEE for practical shall be evaluated for 100 marks and scored marks shall be scaled down to 50 marks (however, based on course type, rubrics shall be decided by the examiners)
- The duration of SEE is 02 hours

Rubrics suggested in Annexure-II of Regulation book

Textbooks

1. Jones, O., Maillardet. R. and Robinson, A. (2014). Introduction to Scientific Programming and Simulation Using R. Chapman & Hall/CRC, The R Series.

References:

1. Michael J. Crawley, "Statistics: An Introduction using R", Second edition, Wiley,2015

Weblinks and Video Lectures (e-Resources):

1. Wickham, H. & Golemund, G. (2018). for Data Science. O'Reilly: New York. Available for free at <http://r4ds.had.co.nz>

Activity Based Learning (Suggested Activities in Class)/ Practical Based learning

- Demonstration of simple projects

MANAGEMENT AND ENTREPRENEURSHIP FOR IT INDUSTRY**(Effective from the academic year 2018 -2019)****SEMESTER – V**

Course Code	18CS51	CIE Marks	40
Number of Contact Hours/Week	2:2:0	SEE Marks	60
Total Number of Contact Hours	40	Exam Hours	03
CREDITS – 03			
Course Learning Objectives: This course (18CS51) will enable students to:			
<ul style="list-style-type: none"> • Explain the principles of management, organization and entrepreneur. • Discuss on planning, staffing, ERP and their importance • Infer the importance of intellectual property rights and relate the institutional support 			
Module – 1			Contact Hours
Introduction - Meaning, nature and characteristics of management, scope and Functional areas of management, goals of management, levels of management, brief overview of evolution of management theories,. Planning- Nature, importance, types of plans, steps in planning, Organizing- nature and purpose, types of Organization, Staffing- meaning, process of recruitment and selection RBT: L1, L2			08
Module – 2			
Directing and controlling- meaning and nature of directing, leadership styles, motivation Theories, Communication- Meaning and importance, Coordination- meaning and importance, Controlling- meaning, steps in controlling, methods of establishing control. RBT: L1, L2			08
Module – 3			
Entrepreneur – meaning of entrepreneur, characteristics of entrepreneurs, classification and types of entrepreneurs, various stages in entrepreneurial process, role of entrepreneurs in economic development, entrepreneurship in India and barriers to entrepreneurship. Identification of business opportunities, market feasibility study, technical feasibility study, financial feasibility study and social feasibility study. RBT: L1, L2			08
Module – 4			
Preparation of project and ERP - meaning of project, project identification, project selection, project report, need and significance of project report, contents, formulation, guidelines by planning commission for project report, Enterprise Resource Planning: Meaning and Importance- ERP and Functional areas of Management – Marketing / Sales- Supply Chain Management – Finance and Accounting – Human Resources – Types of reports and methods of report generation RBT: L1, L2			08
Module – 5			
Micro and Small Enterprises: Definition of micro and small enterprises, characteristics and advantages of micro and small enterprises, steps in establishing micro and small enterprises, Government of India industrial policy 2007 on micro and small enterprises, case study (Microsoft), Case study(Captain G R Gopinath),case study (N R Narayana Murthy & Infosys), Institutional support: MSME-DI, NSIC, SIDBI, KIADB, KSSIDC, TECSOK, KSFC, DIC and District level single window agency, Introduction to IPR. RBT: L1, L2			08

Course outcomes: The students should be able to:
<ul style="list-style-type: none"> • Define management, organization, entrepreneur, planning, staffing, ERP and outline their importance in entrepreneurship • Utilize the resources available effectively through ERP • Make use of IPRs and institutional support in entrepreneurship
Question Paper Pattern:
<ul style="list-style-type: none"> • 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.
Textbooks:
<ol style="list-style-type: none"> 1. Principles of Management -P. C. Tripathi, P. N. Reddy; Tata McGraw Hill, 4th / 6th Edition, 2010. 2. Dynamics of Entrepreneurial Development & Management -Vasant Desai Himalaya Publishing House. 3. Entrepreneurship Development -Small Business Enterprises -Poornima M Charantimath Pearson Education – 2006. 4. Management and Entrepreneurship - Kanishka Bedi- Oxford University Press-2017
Reference Books:
<ol style="list-style-type: none"> 1. Management Fundamentals -Concepts, Application, Skill Development Robert Lusier – Thomson. 2. Entrepreneurship Development -S S Khanka -S Chand & Co. 3. Management -Stephen Robbins -Pearson Education /PHI -17th Edition, 2003

COMPUTER NETWORKS AND SECURITY (Effective from the academic year 2018 -2019) SEMESTER – V			
Course Code	18CS52	CIE Marks	40
Number of Contact Hours/Week	3:2:0	SEE Marks	60
Total Number of Contact Hours	50	Exam Hours	03
CREDITS –4			
Course Learning Objectives: This course (18CS52) will enable students to:			
<ul style="list-style-type: none"> • Demonstration of application layer protocols • Discuss transport layer services and understand UDP and TCP protocols • Explain routers, IP and Routing Algorithms in network layer • Disseminate the Wireless and Mobile Networks covering IEEE 802.11 Standard • Illustrate concepts of Multimedia Networking, Security and Network Management 			
Module 1			Contact Hours

<p>Application Layer: Principles of Network Applications: Network Application Architectures, Processes Communicating, Transport Services Available to Applications, Transport Services Provided by the Internet, Application-Layer Protocols. The Web and HTTP: Overview of HTTP, Non-persistent and Persistent Connections, HTTP Message Format, User-Server Interaction: Cookies, Web Caching, The Conditional GET, File Transfer: FTP Commands & Replies, Electronic Mail in the Internet: SMTP, Comparison with HTTP, Mail Message Format, Mail Access Protocols, DNS; The Internet's Directory Service: Services Provided by DNS, Overview of How DNS Works, DNS Records and Messages, Peer-to-Peer Applications: P2P File Distribution, Distributed Hash Tables, Socket Programming: creating Network Applications: Socket Programming with UDP, Socket Programming with TCP.</p> <p>T1: Chap 2 RBT: L1, L2, L3</p>	10
Module 2	
<p>Transport Layer : Introduction and Transport-Layer Services: Relationship Between Transport and Network Layers, Overview of the Transport Layer in the Internet, Multiplexing and Demultiplexing: Connectionless Transport: UDP,UDP Segment Structure, UDP Checksum, Principles of Reliable Data Transfer: Building a Reliable Data Transfer Protocol, Pipelined Reliable Data Transfer Protocols, Go-Back-N, Selective repeat, Connection-Oriented Transport TCP: The TCP Connection, TCP Segment Structure, RoundTrip Time Estimation and Timeout, Reliable Data Transfer, Flow Control, TCP Connection Management, Principles of Congestion Control: The Causes and the Costs of Congestion, Approaches to Congestion Control, Network-assisted congestion-control example, ATM ABR Congestion control, TCP Congestion Control: Fairness.</p> <p>T1: Chap 3 RBT: L1, L2, L3</p>	10
Module 3	
<p>The Network layer: What's Inside a Router?: Input Processing, Switching, Output Processing, Where Does Queuing Occur? Routing control plane, IPv6,A Brief foray into IP Security, Routing Algorithms: The Link-State (LS) Routing Algorithm, The Distance-Vector (DV) Routing Algorithm, Hierarchical Routing, Routing in the Internet, Intra-AS Routing in the Internet: RIP, Intra-AS Routing in the Internet: OSPF, Inter/AS Routing: BGP, Broadcast Routing Algorithms and Multicast.</p> <p>T1: Chap 4: 4.3-4.7 RBT: L1, L2, L3</p>	10
Module 4	
<p>Network Security:Overview of Network Security:Elements of Network Security , Classification of Network Attacks ,Security Methods ,Symmetric-Key Cryptography :Data Encryption Standard (DES),Advanced Encryption Standard (AES) , Public-Key Cryptography :RSA Algorithm ,Diffie-Hellman Key-Exchange Protocol , Authentication :Hash Function , Secure Hash Algorithm (SHA) , Digital Signatures , Firewalls and Packet Filtering ,Packet Filtering , Proxy Server .</p> <p>Textbook2: Chapter 10 RBT: L1, L2, L3</p>	10
Module 5	

Multimedia Networking: Properties of video, properties of Audio, Types of multimedia Network Applications, Streaming stored video: UDP Streaming, HTTP Streaming, Adaptive streaming and DASH, content distribution Networks Voice-over-IP :Limitations of the Best-Effort IP Service ,Removing Jitter at the Receiver for Audio ,Recovering from Packet Loss Protocols for Real-Time Conversational Applications , RTP , SIP Textbook11: Chap 7 RBT: L1, L2, L3	10
Course Outcomes: The student will be able to :	
<ul style="list-style-type: none"> • Explain principles of application layer protocols • Recognize transport layer services and infer UDP and TCP protocols • Classify routers, IP and Routing Algorithms in network layer • Understand the Wireless and Mobile Networks covering IEEE 802.11 Standard • Describe Multimedia Networking and Network Management 	
Question Paper Pattern:	
<ul style="list-style-type: none"> • 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. 	
Textbooks:	
<ol style="list-style-type: none"> 1. James F Kurose and Keith W Ross, Computer Networking, A Top-Down Approach, Sixth edition, Pearson,2017 . 2. Nader F Mir, Computer and Communication Networks, 2nd Edition, Pearson, 2014. 	
Reference Books:	
<ol style="list-style-type: none"> 1. Behrouz A Forouzan, Data and Communications and Networking, Fifth Edition, McGraw Hill, Indian Edition 2. Larry L Peterson and Bruce S Davie, Computer Networks, fifth edition, ELSEVIER 3. Andrew S Tanenbaum, Computer Networks, fifth edition, Pearson 4. Mayank Dave, Computer Networks, Second edition, Cengage Learning 	

DATABASE MANAGEMENT SYSTEM			
(Effective from the academic year 2018 -2019)			
SEMESTER – V			
Course Code	18CS53	CIE Marks	40
Number of Contact Hours/Week	3:2:0	SEE Marks	60
Total Number of Contact Hours	50	Exam Hours	03
CREDITS –4			
Course Learning Objectives: This course (18CS53) will enable students to:			
<ul style="list-style-type: none"> • Provide a strong foundation in database concepts, technology, and practice. • Practice SQL programming through a variety of database problems. □ • Demonstrate the use of concurrency and transactions in database □ • Design and build database applications for real world problems. 			
Module 1		Contact Hours	

<p>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, The Database System environment. Conceptual Data Modelling using Entities and Relationships: Entity types, Entity sets, attributes, roles, and structural constraints, Weak entity types, ER diagrams, examples, Specialization and Generalization. Textbook 1:Ch 1.1 to 1.8, 2.1 to 2.6, 3.1 to 3.10 RBT: L1, L2, L3</p>	10
Module 2	
<p>Relational Model: Relational Model Concepts, Relational Model Constraints and relational database schemas, Update operations, transactions, and dealing with constraint violations. Relational Algebra: Unary and Binary relational operations, additional relational operations (aggregate, grouping, etc.) Examples of Queries in relational algebra. 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. Textbook 1: Ch4.1 to 4.5, 5.1 to 5.3, 6.1 to 6.5, 8.1; Textbook 2: 3.5 RBT: L1, L2, L3</p>	10
Module 3	
<p>SQL : Advances Queries: More complex SQL retrieval queries, Specifying constraints as assertions and action triggers, Views in SQL, Schema change statements in SQL. Database Application Development: Accessing databases from applications, An introduction to JDBC, JDBC classes and interfaces, SQLJ, Stored procedures, Case study: The internet Bookshop. Internet Applications: The three-Tier application architecture, The presentation layer, The Middle Tier Textbook 1: Ch7.1 to 7.4; Textbook 2: 6.1 to 6.6, 7.5 to 7.7. RBT: L1, L2, L3</p>	10
Module 4	
<p>Normalization: Database Design Theory – Introduction to Normalization using Functional and Multivalued Dependencies: Informal design guidelines for relation schema, Functional Dependencies, Normal Forms based on Primary Keys, Second and Third Normal Forms, Boyce-Codd Normal Form, Multivalued Dependency and Fourth Normal Form, Join Dependencies and Fifth Normal Form. Normalization Algorithms: Inference Rules, Equivalence, and Minimal Cover, Properties of Relational Decompositions, Algorithms for Relational Database Schema Design, Nulls, Dangling tuples, and alternate Relational</p>	10
<p>Designs, Further discussion of Multivalued dependencies and 4NF, Other dependencies and Normal Forms Textbook 1: Ch14.1 to 14.7, 15.1 to 15.6 RBT: L1, L2, L3</p>	
Module 5	

<p>Transaction Processing: Introduction to Transaction Processing, Transaction and System concepts, Desirable properties of Transactions, Characterizing schedules based on recoverability, Characterizing schedules based on Serializability, Transaction support in SQL. 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. Introduction to Database Recovery Protocols: Recovery Concepts, NO-UNDO/REDO recovery based on Deferred update, Recovery techniques based on immediate update, Shadow paging, Database backup and recovery from catastrophic failures Textbook 1: 20.1 to 20.6, 21.1 to 21.7, 22.1 to 22.4, 22.7. RBT: L1, L2, L3</p>	10
<p>Course Outcomes: The student will be able to :</p>	
<ul style="list-style-type: none"> • Identify, analyze and define database objects, enforce integrity constraints on a database using RDBMS. • Use Structured Query Language (SQL) for database manipulation. □ Design and build simple database systems • Develop application to interact with databases. 	
<p>Question Paper Pattern:</p>	
<ul style="list-style-type: none"> • 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. 	
<p>Textbooks:</p>	
<ol style="list-style-type: none"> 1. Fundamentals of Database Systems, Ramez Elmasri and Shamkant B. Navathe, 7th Edition, 2017, Pearson. 2. Database management systems, Ramakrishnan, and Gehrke, 3rd Edition, 2014, McGraw Hill 	
<p>Reference Books:</p>	
<ol style="list-style-type: none"> 1. Silberschatz Korth and Sudharshan, Database System Concepts, 6th Edition, Mc-GrawHill, 2013. 2. Coronel, Morris, and Rob, Database Principles Fundamentals of Design, Implementation and Management, Cengage Learning 2012. 	

<p>AUTOMATA THEORY AND COMPUTABILITY (Effective from the academic year 2018 -2019) SEMESTER – V</p>			
Course Code	18CS54	CIE Marks	40
Number of Contact Hours/Week	3:0:0	SEE Marks	60
Total Number of Contact Hours	40	Exam Hours	03
<p>CREDITS –3</p>			
<p>Course Learning Objectives: This course (18CS54) will enable students to:</p>			
<ul style="list-style-type: none"> • 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 			

Module 1	Contact Hours
<p>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, Canonical form of Regular languages, Finite State Transducers, Bidirectional Transducers.</p> <p>Textbook 1: Ch 1,2, 3,4, 5.1 to 5.10 RBT: L1, L2</p>	08
<p>Module 2</p> <p>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.</p> <p>Textbook 1: Ch 6, 7, 8: 6.1 to 6.4, 7.1, 7.2, 8.1 to 8.4 RBT: L1, L2, L3</p>	08
<p>Module 3</p> <p>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.</p> <p>Textbook 1: Ch 11, 12: 11.1 to 11.8, 12.1, 12.2, 12.4, 12.5, 12.6 RBT: L1, L2, L3</p>	08
<p>Module 4</p> <p>Algorithms and Decision Procedures for CFLs: Decidable questions, Un-decidable questions. Turing Machine: Turing machine model, Representation, Language acceptability by TM, design of TM, Techniques for TM construction. Variants of Turing Machines (TM), The model of Linear Bounded automata.</p> <p>Textbook 1: Ch 14: 14.1, 14.2, Textbook 2: Ch 9.1 to 9.8 RBT: L1, L2, L3</p>	08
<p>Module 5</p> <p>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, ChurchTuring thesis. Applications: G.1 Defining syntax of programming language, Appendix J: Security</p> <p>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</p>	08
<p>Course Outcomes: The student will be able to :</p>	

<ul style="list-style-type: none"> • 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.
Question Paper Pattern:
<ul style="list-style-type: none"> • 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.
Textbooks:
<ol style="list-style-type: none"> 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:
<ol style="list-style-type: none"> 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.
Faculty can utilize open source tools (like JFLAP) to make teaching and learning more interactive.

APPLICATION DEVELOPMENT USING PYTHON			
[(Effective from the academic year 2018 -2019)			
SEMESTER – V			
Course Code	18CS55	IA Marks	40
Number of Lecture Hours/Week	03	Exam Marks	60
Total Number of Lecture Hours	40	Exam Hours	03
CREDITS – 03			
Course Learning Objectives: This course (18CS55) will enable students to			
<ul style="list-style-type: none"> • 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. 			

Module – 1	Teaching Hours
<p>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</p>	08
Module – 2	
<p>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</p>	08
Module – 3	
<p>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 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</p>	08
RBT: L1, L2, L3	
Module – 4	
<p>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</p>	08

Module – 5	
<p>Web Scraping, Project: MAPIT.PY with the webbrowser Module, Downloading Files from 08 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</p> <p>Textbook 1: Chapters 11 – 14 RBT: L1, L2, L3</p>	
Course Outcomes: After studying this course, students will be able to	
<ul style="list-style-type: none"> • Demonstrate proficiency in handling of loops and creation of functions. • Identify the methods to create and manipulate lists, tuples and dictionaries. • Discover the commonly used operations involving regular expressions and file system. • Interpret the concepts of Object-Oriented Programming as used in Python. • Determine the need for scraping websites and working with CSV, JSON and other file formats. 	
Question paper pattern:	
<ul style="list-style-type: none"> • 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 Books:	
<ol style="list-style-type: none"> 1. Al Sweigart, “Automate the Boring Stuff with Python”, 1st Edition, 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”, 2nd Edition, Green Tea Press, 2015. (Available under CC-BY-NC license at http://greenteapress.com/thinkpython2/thinkpython2.pdf) (Chapters 13, 15, 16, 17, 18) (Download pdf/html files from the above links) 	
Reference Books:	
<ol style="list-style-type: none"> 1. Gowrishankar S, Veena A, “Introduction to Python Programming”, 1st Edition, CRC Press/Taylor & Francis, 2018. ISBN-13: 978-0815394372 2. Jake VanderPlas, “Python Data Science Handbook: Essential Tools for Working with Data”, 1st Edition, O’Reilly Media, 2016. ISBN-13: 978-1491912058 3. Charles Dierbach, “Introduction to Computer Science Using Python”, 1st Edition, Wiley India Pvt Ltd, 2015. ISBN-13: 978-8126556014 4. Wesley J Chun, “Core Python Applications Programming”, 3rd Edition, Pearson Education India, 2015. ISBN-13: 978-9332555365 	

UNIX PROGRAMMING
(Effective from the academic year 2018 -2019)
SEMESTER – V

Course Code	18CS56	CIE Marks	40
Number of Contact Hours/Week	3:0:0	SEE Marks	60
Total Number of Contact Hours	40	Exam Hours	03
CREDITS – 3			
Course Learning Objectives: This course (18CS56) will enable students to			
<ul style="list-style-type: none"> • Interpret the features of UNIX and basic commands. □ • Demonstrate different UNIX files and permissions □ • Implement shell programs. • Explain UNIX process, IPC and signals. 			
Module 1			Contact Hours
<p>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.</p> <p>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.</p> <p>RBT: L1, L2</p>			08
Module 2			
<p>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.</p> <p>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.</p> <p>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.</p> <p>RBT: L1, L2</p>			08
Module 3			

<p>UNIX File APIs: General File APIs, File and Record Locking, Directory File APIs, Device File APIs, FIFO File APIs, Symbolic Link File APIs.</p> <p>UNIX Processes and Process Control:</p> <p>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.</p> <p>Process Control: Introduction, Process Identifiers, fork, vfork, exit, wait, waitpid, wait3, wait4 Functions, Race Conditions, exec Functions RBT:</p> <p>L1, L2, L3</p>	08
Module 4	
<p>Changing User IDs and Group IDs, Interpreter Files, system Function, Process Accounting, User Identification, Process Times, I/O Redirection.</p> <p>Overview of IPC Methods, Pipes, popen, pclose Functions, Coprocesses, FIFOs, System V IPC, Message Queues, Semaphores.</p> <p>Shared Memory, Client-Server Properties, Stream Pipes, Passing File Descriptors, An Open Server-Version 1, Client-Server Connection Functions.</p> <p>RBT: L1, L2, L3</p>	08
Module 5	
<p>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.1b Timers. Daemon Processes: Introduction, Daemon Characteristics, Coding Rules, Error Logging, Client-Server Model.</p> <p>RBT: L1, L2, L3</p>	08
Course Outcomes: The student will be able to :	
<ul style="list-style-type: none"> • 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. 	
Question Paper Pattern:	
<ul style="list-style-type: none"> • 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. 	
Textbooks:	
<ol style="list-style-type: none"> 1. Sumitabha Das., Unix Concepts and Applications., 4thEdition., Tata McGraw Hill (Chapter 1,2 ,3,4,5,6,8,13,14) 2. W. Richard Stevens: Advanced Programming in the UNIX Environment, 2nd Edition, Pearson Education, 2005 (Chapter 3,7,8,10,13,15) 3. Unix System Programming Using C++ - Terrence Chan, PHI, 1999. (Chapter 7,8,9,10) 	
Reference Books:	
<ol style="list-style-type: none"> 1. M.G. Venkatesh Murthy: UNIX & Shell Programming, Pearson Education. 2. Richard Blum , Christine Bresnahan : Linux Command Line and Shell Scripting Bible, 2ndEdition, Wiley,2014. 	
Faculty can utilize open source tools to make teaching and learning more interactive.	

(Effective from the academic year 2018 -2019)

SEMESTER – V

Course Code	18CSL57	CIE Marks	40
Number of Contact Hours/Week	0:2:2	SEE Marks	60
Total Number of Lab Contact Hours	36	Exam Hours	03
Credits – 2			

Course Learning Objectives: This course (18CSL57) will enable students to:

- Demonstrate operation of network and its management commands □ Simulate and demonstrate the performance of GSM and CDMA □ Implement data link layer and transport layer protocols.

Descriptions (if any):

- For the experiments below modify the topology and parameters set for the experiment and take multiple rounds of reading and analyze the results available in log files. Plot necessary graphs and conclude. Use NS2/NS3.
- **Installation procedure of the required software must be demonstrated, carried out in groups and documented in the journal.**

Programs List:

PART A

1.	Implement three nodes point – to – point network with duplex links between them. Set the queue size, vary the bandwidth and find the number of packets dropped.
2.	Implement transmission of ping messages/trace route over a network topology consisting of 6 nodes and find the number of packets dropped due to congestion.
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

PART B (Implement the following in Java)

7.	Write a program for error detecting code using CRC-CCITT (16- bits).
8.	Write a program to find the shortest path between vertices using bellman-ford algorithm.
9.	Using TCP/IP sockets, write a client – server program to make the client send the file name and to make the server send back the contents of the requested file if present.
10.	Write a program on datagram socket for client/server to display the messages on client side, typed at the server side.
11.	Write a program for simple RSA algorithm to encrypt and decrypt the data.
12.	Write a program for congestion control using leaky bucket algorithm.

Laboratory Outcomes: The student should be able to:

- Analyze and Compare various networking protocols.
- Demonstrate the working of different concepts of networking.
- Implement, analyze and evaluate networking protocols in NS2 / NS3 and JAVA programming language

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 accordance with university regulations*)
 - i) For laboratories having only one part – Procedure + Execution + Viva-Voce: $15+70+15 = 100$ Marks
 - j) For laboratories having PART A and PART B
 - i. Part A – Procedure + Execution + Viva = $6 + 28 + 6 = 40$ Marks ii.
 - Part B – Procedure + Execution + Viva = $9 + 42 + 9 = 60$ Marks

DBMS LABORATORY WITH MINI PROJECT
(Effective from the academic year 2018 -2019)
SEMESTER – V

Course Code	18CSL58	CIE Marks	40
Number of Contact Hours/Week	0:2:2	SEE Marks	60
Total Number of Lab Contact Hours	36	Exam Hours	03

Credits – 2

Course Learning Objectives: This course (18CSL58) will enable students to:

- Foundation knowledge in database concepts, technology and practice to groom students into well-informed database application developers.
- Strong practice in SQL programming through a variety of database problems.
- Develop database applications using front-end tools and back-end DBMS.

Descriptions (if any):

PART-A: SQL Programming (Max. Exam Mks. 50)

- Design, develop, and implement the specified queries for the following problems using Oracle, MySQL, MS SQL Server, or any other DBMS under LINUX/Windows environment.
- Create Schema and insert at least 5 records for each table. Add appropriate database constraints.

PART-B: Mini Project (Max. Exam Mks. 30)

- Use Java, C#, PHP, Python, or any other similar front-end tool. All applications must be demonstrated on desktop/laptop as a stand-alone or web based application (Mobile apps on Android/IOS are not permitted.)

Installation procedure of the required software must be demonstrated, carried out in groups and documented in the journal.

Programs List:

PART A

- | | |
|----|---|
| 1. | <p>Consider the following schema for a Library Database:
 BOOK(Book_id, Title, Publisher_Name, Pub_Year)
 BOOK_AUTHORS(Book_id, Author_Name)
 PUBLISHER(Name, Address, Phone)
 BOOK_COPIES(Book_id, Programme_id, No-of_Copies)
 BOOK_LENDING(Book_id, Programme_id, Card_No, Date_Out, Due_Date)
 LIBRARY_PROGRAMME(Programme_id, Programme_Name, Address)</p> <p>Write SQL queries to</p> <ol style="list-style-type: none"> 1. Retrieve details of all books in the library – id, title, name of publisher, authors, number of copies in each Programme, etc. 2. Get the particulars of borrowers who have borrowed more than 3 books, but from Jan 2017 to Jun 2017. 3. Delete a book in BOOK table. Update the contents of other tables to reflect this data manipulation operation. 4. Partition the BOOK table based on year of publication. Demonstrate its working with a simple query. 5. Create a view of all books and its number of copies that are currently available in the Library. |
|----|---|

2.	<p>Consider the following schema for Order Database: SALESMAN(<u>Salesman_id</u>, Name, City, Commission) CUSTOMER(<u>Customer_id</u>, Cust_Name, City, Grade, Salesman_id) ORDERS(<u>Ord_No</u>, Purchase_Amt, Ord_Date, Customer_id, Salesman_id)</p> <p>Write SQL queries to</p> <ol style="list-style-type: none"> Count the customers with grades above Bangalore's average.
	<ol style="list-style-type: none"> Find the name and numbers of all salesman who had more than one customer. List all the salesman and indicate those who have and don't have customers in their cities (Use UNION operation.) Create a view that finds the salesman who has the customer with the highest order of a day. Demonstrate the DELETE operation by removing salesman with id 1000. All his orders must also be deleted.
3.	<p>Consider the schema for Movie Database: ACTOR(<u>Act_id</u>, Act_Name, Act_Gender) DIRECTOR(<u>Dir_id</u>, Dir_Name, Dir_Phone) MOVIES(<u>Mov_id</u>, Mov_Title, Mov_Year, Mov_Lang, Dir_id) MOVIE_CAST(<u>Act_id</u>, <u>Mov_id</u>, Role) RATING(<u>Mov_id</u>, Rev_Stars)</p> <p>Write SQL queries to</p> <ol style="list-style-type: none"> List the titles of all movies directed by „Hitchcock“. Find the movie names where one or more actors acted in two or more movies. List all actors who acted in a movie before 2000 and also in a movie after 2015 (use JOIN operation). Find the title of movies and number of stars for each movie that has at least one rating and find the highest number of stars that movie received. Sort the result by movie title. Update rating of all movies directed by „Steven Spielberg“ to 5.
4.	<p>Consider the schema for College Database: STUDENT(<u>USN</u>, SName, Address, Phone, Gender) SEMSEC(<u>SSID</u>, Sem, Sec) CLASS(<u>USN</u>, <u>SSID</u>) COURSE(<u>Subcode</u>, Title, Sem, Credits) IAMARKS(<u>USN</u>, <u>Subcode</u>, <u>SSID</u>, Test1, Test2, Test3, FinalIA)</p> <p>Write SQL queries to</p> <ol style="list-style-type: none"> List all the student details studying in fourth semester „C“ section. Compute the total number of male and female students in each semester and in each section. Create a view of Test1 marks of student USN „1BI15CS101“ in all Courses. Calculate the FinalIA (average of best two test marks) and update the corresponding table for all students. Categorize students based on the following criterion: If FinalIA = 17 to 20 then CAT = „Outstanding“ If FinalIA = 12 to 16 then CAT = „Average“ If FinalIA < 12 then CAT = „Weak“ Give these details only for 8th semester A, B, and C section students.

5.	<p>Consider the schema for Company Database: EMPLOYEE(SSN, Name, Address, Sex, Salary, SuperSSN, DNo) DEPARTMENT(DNo, DName, MgrSSN, MgrStartDate) DLOCATION(DNo, DLoc) PROJECT(PNo, PName, PLocation, DNo) WORKS_ON(SSN, PNo, Hours) Write SQL queries to</p> <ol style="list-style-type: none"> 1. Make a list of all project numbers for projects that involve an employee whose last name is „Scott“, either as a worker or as a manager of the department that controls the project. 2. Show the resulting salaries if every employee working on the „IoT“ project is
	<ol style="list-style-type: none"> 3. given a 10 percent raise. 3. Find the sum of the salaries of all employees of the „Accounts“ department, as well as the maximum salary, the minimum salary, and the average salary in this department 4. Retrieve the name of each employee who works on all the projects controlled by department number 5 (use NOT EXISTS operator). 5. For each department that has more than five employees, retrieve the department number and the number of its employees who are making more than Rs. 6,00,000.
PART B: Mini Project	
<input type="checkbox"/>	For any problem selected
<input type="checkbox"/>	Make sure that the application should have five or more tables
<input type="checkbox"/>	Indicative areas include; health care
Laboratory Outcomes: The student should be able to:	
<ul style="list-style-type: none"> • Create, Update and query on the database. • Demonstrate the working of different concepts of DBMS • Implement, analyze and evaluate the project developed for an application. 	
Conduct of Practical Examination:	
<ul style="list-style-type: none"> • Experiment distribution <ul style="list-style-type: none"> ○ 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 (<i>Courseed to change in accordance with university regulations</i>) <ol style="list-style-type: none"> k) For laboratories having only one part – Procedure + Execution + Viva-Voce: 15+70+15 = 100 Marks l) For laboratories having PART A and PART B <ol style="list-style-type: none"> i. Part A – Procedure + Execution + Viva = 6 + 28 + 6 = 40 Marks ii. Part B – Procedure + Execution + Viva = 9 + 42 + 9 = 60 Marks 	

B. E. COMMON TO ALL PROGRAMMES
Choice Based Credit System (CBCS) and Outcome Based Education (OBE) SEMESTER
- V

ENVIRONMENTAL STUDIES

Course Code	18CIV59	CIE Marks	40
Teaching Hours / Week (L:T:P)	(1:0:0)	SEE Marks	60
Credits	01	Exam Hours	02

Module - 1

Ecosystems (Structure and Function): Forest, Desert, Wetlands, Riverine, Oceanic and Lake.
Biodiversity: Types, Value; Hot-spots; Threats and Conservation of biodiversity, Forest Wealth, and Deforestation.

Module - 2

Advances in Energy Systems (Merits, Demerits, Global Status and Applications): Hydrogen, Solar, OTEC, Tidal and Wind.
Natural Resource Management (Concept and case-studies): Disaster Management, Sustainable Mining, Cloud Seeding, and Carbon Trading.

Module - 3

Environmental Pollution (Sources, Impacts, Corrective and Preventive measures, Relevant Environmental Acts, Case-studies): Surface and Ground Water Pollution; Noise pollution; Soil Pollution and Air Pollution. **Waste Management & Public Health Aspects:** Bio-medical Wastes; Solid waste; Hazardous wastes; E-wastes; Industrial and Municipal Sludge.

Module - 4

Global Environmental Concerns (Concept, policies and case-studies): Ground water depletion/recharging, Climate Change; Acid Rain; Ozone Depletion; Radon and Fluoride problem in drinking water; Resettlement and rehabilitation of people, Environmental Toxicology.

Module - 5

Latest Developments in Environmental Pollution Mitigation Tools (Concept and Applications): G.I.S. & Remote Sensing, Environment Impact Assessment, Environmental Management Systems, ISO14001; Environmental Stewardship- NGOs.

Field work: Visit to an Environmental Engineering Laboratory or Green Building or Water Treatment Plant or Waste water treatment Plant; ought to be Followed by understanding of process and its brief documentation.

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

- CO1: Understand the principles of ecology and environmental issues that apply to air, land, and water issues on a global scale,
- CO2: Develop critical thinking and/or observation skills, and apply them to the analysis of a problem or question related to the environment.
- CO3: Demonstrate ecology knowledge of a complex relationship between biotic and abiotic components.
- CO4: Apply their ecological knowledge to illustrate and graph a problem and describe the realities that managers face when dealing with complex issues.

Question paper pattern:

- The Question paper will have 100 objective questions.
- Each question will be for 01 marks
- Student will have to answer all the questions in an OMR Sheet.
- The Duration of Exam will be 2 hours.

Sl. No.	Title of the Book	Name of the Author/s	Name of the Publisher	Edition and Year
---------	-------------------	----------------------	-----------------------	------------------

Textbook/s

1	Environmental Studies	Benny Joseph	Tata Mc Graw – Hill.	2 nd Edition, 2012
2.	Environmental Studies	S M Prakash	Pristine Publishing House, Mangalore	3 rd Edition 2018
3	Environmental Studies – From Crisis to Cure	R Rajagopalan	Oxford Publisher	2005
Reference Books				
1	Principals of Environmental Science and Engineering	Raman Sivakumar	Cengage learning, Singapur.	2 nd Edition, 2005
2	Environmental Science – working with the Earth	G.Tyler Miller Jr.	Thomson Brooks /Cole,	11 th Edition, 2006
3	Text Book of Environmental and Ecology	Pratiba Sing, Anoop Singh & Piyush Malaviya	Acme Learning Pvt. Ltd. New Delhi.	1 st Edition

SYSTEM SOFTWARE AND COMPILERS
(Effective from the academic year 2018 -2019) SEMESTER
- VI

Course Code	18CS61	CIE Marks	40
Number of Contact Hours/Week	3:2:0	SEE Marks	60
Total Number of Contact Hours	50	Exam Hours	03
CREDITS –4			
Course Learning Objectives: This course (18CS61) will enable students to:			
<ul style="list-style-type: none"> • 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 			
Module 1			Contact Hours
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
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
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
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 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			10
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			10
Course Outcomes: 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

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.

Textbooks:

1. System Software by Leland. L. Beck, D Manjula, 3rd edition, 2012
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. Dhamdhare TMG
4. Compiler Design, K Muneeswaran, Oxford University Press 2013.

COMPUTER GRAPHICS AND VISUALIZATION
(Effective from the academic year 2018 -2019) SEMESTER
- VI

Course Code	18CS62	CIE Marks	40
Number of Contact Hours/Week	3:2:0	SEE Marks	60
Total Number of Contact Hours	50	Exam Hours	03
CREDITS –4			
Course Learning Objectives: This course (18CS62) will enable students to:			
<ul style="list-style-type: none"> • Explain hardware, software and OpenGL Graphics Primitives. • Illustrate interactive computer graphic using the OpenGL. • Design and implementation of algorithms for 2D graphics Primitives and attributes. • Demonstrate Geometric transformations, viewing on both 2D and 3D objects. • Infer the representation of curves, surfaces, Color and Illumination models 			
Module 1			Contact Hours
<p>Overview: Computer Graphics and OpenGL: Computer Graphics: Basics of computer graphics, Application of Computer Graphics, Video Display Devices: Random Scan and Raster Scan displays, graphics software. OpenGL: Introduction to OpenGL ,coordinate reference frames, specifying two-dimensional world coordinate reference frames in OpenGL, OpenGL point functions, OpenGL line functions, point attributes, line attributes, curve attributes, OpenGL point attribute functions, OpenGL line attribute functions, Line drawing algorithms(DDA, Bresenham"s), circle generation algorithms (Bresenham"s). Text-1:Chapter -1: 1-1 to 1-9, 2-1(page 39 to 41),2.8,2.9,3-1 to 3-5,3-9,3-20 RBT: L1, L2, L3</p>			10
Module 2			
<p>Fill area Primitives, 2D Geometric Transformations and 2D viewing: Fill area Primitives: Polygon fill-areas, OpenGL polygon fill area functions, fill area attributes, general scan line polygon fill algorithm, OpenGL fill-area attribute functions. 2DGeometric Transformations: Basic 2D Geometric Transformations, matrix representations and homogeneous coordinates. Inverse transformations, 2DComposite transformations, other 2D transformations, raster methods for geometric transformations, OpenGL raster transformations, OpenGL geometric transformations function, 2D viewing: 2D viewing pipeline, OpenGL 2D viewing functions. Text-1:Chapter 3-14 to 3-16,4-9,4-10,4-14,5-1 to 5-7,5-17,6-1,6-4 RBT: L1, L2, L3</p>			10
Module 3			
<p>Clipping,3D Geometric Transformations, Color and Illumination Models: Clipping: clipping window, normalization and viewport transformations, clipping algorithms,2D point clipping, 2D line clipping algorithms: cohen-sutherland line clipping only -polygon fill area clipping: Sutherland-Hodgeman polygon clipping algorithm only.3DGeometric Transformations: 3D translation, rotation, scaling, composite 3D transformations, other 3D transformations, affine transformations, OpenGL geometric transformations functions. Color Models: Properties of light, color models, RGB and CMY color models. Illumination Models: Light sources, basic illumination models-Ambient light, diffuse reflection, specular and phong model, Corresponding openGL functions. Text-1:Chapter :6-2 to 6-08 (Excluding 6-4),5-9 to 5-17(Excluding 5-15),12-1,12-2,124,12-6,10-1,10-3 RBT: L1, L2, L3</p>			10

Module 4	
3D Viewing and Visible Surface Detection: 3DViewing:3D viewing concepts, 3D viewing pipeline, 3D viewing coordinate parameters , Transformation from world to viewing coordinates, Projection transformation, orthogonal projections, perspective projections, The viewport transformation and 3D screen coordinates. OpenGL 3D viewing functions. Visible Surface Detection Methods: Classification of visible surface Detection algorithms, depth buffer method only and OpenGL visibility detection functions. Text-1:Chapter: 7-1 to 7-10(Excluding 7-7), 9-1,9-3, 9-14 RBT: L1, L2, L3	10
Module 5	
Input& interaction, Curves and Computer Animation: Input and Interaction: Input devices, clients and servers, Display Lists, Display Lists and Modeling, Programming Event Driven Input, Menus Picking, Building Interactive Models, Animating Interactive programs, Design of Interactive programs, Logic operations .Curved surfaces, quadric surfaces, OpenGL Quadric-Surface and Cubic-Surface Functions, Bezier Spline Curves, Bezier surfaces, OpenGL curve functions. Corresponding openGL functions. Text-1:Chapter :8-3 to 8-6 (Excluding 8-5),8-9,8-10,8-11,3-8,8-18,13-11,3-2,13-3,134,13-10 Text-2:Chapter 3: 3-1 to 3.11: Input& interaction RBT: L1, L2, L3	10
Course Outcomes: The student will be able to :	
<ul style="list-style-type: none"> • Design and implement algorithms for 2D graphics primitives and attributes. • Illustrate Geometric transformations on both 2D and 3D objects. • Apply concepts of clipping and visible surface detection in 2D and 3D viewing, and Illumination Models. • Decide suitable hardware and software for developing graphics packages using OpenGL. 	
Question Paper Pattern:	
<ul style="list-style-type: none"> • 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. 	
Textbooks:	
<ol style="list-style-type: none"> 1. Donald Hearn & Pauline Baker: Computer Graphics with OpenGL Version,3rd / 4th Edition, Pearson Education,2011 2. Edward Angel: Interactive Computer Graphics- A Top Down approach with OpenGL, 5th edition. Pearson Education, 2008 	
Reference Books:	
<ol style="list-style-type: none"> 1. James D Foley, Andries Van Dam, Steven K Feiner, John F Huges Computer graphics with OpenGL: pearson education 2. Xiang, Plastock : Computer Graphics , sham"s outline series, 2nd edition, TMG. 3. Kelvin Sung, Peter Shirley, steven Baer : Interactive Computer Graphics, concepts and applications, Cengage Learning 4. M M Raikar & Shreedhara K S Computer Graphics using OpenGL, Cengage publication 	

Course Code	18CS63	CIE Marks	40
Number of Contact Hours/Week	3:2:0	SEE Marks	60
Total Number of Contact Hours	50	Exam Hours	03
CREDITS –4			
Course Learning Objectives: This course (18CS63) will enable students to:			
<ul style="list-style-type: none"> • Illustrate the Semantic Structure of HTML and CSS • Compose forms and tables using HTML and CSS • Design Client-Side programs using JavaScript and Server-Side programs using PHP • Infer Object Oriented Programming capabilities of PHP • Examine JavaScript frameworks such as jQuery and Backbone 			
Module 1			Contact Hours
Introduction to HTML, What is HTML and Where did it come from?, HTML Syntax, Semantic Markup, Structure of HTML Documents, Quick Tour of HTML Elements, HTML5 Semantic Structure Elements, Introduction to CSS, What is CSS, CSS Syntax, Location of Styles, Selectors, The Cascade: How Styles Interact, The Box Model, CSS Text Styling. Textbook 1: Ch. 2, 3 RBT: L1, L2, L3			10
Module 2			
HTML Tables and Forms, Introducing Tables, Styling Tables, Introducing Forms, Form Control Elements, Table and Form Accessibility, Microformats, Advanced CSS: Layout, Normal Flow, Positioning Elements, Floating Elements, Constructing Multicolumn Layouts, Approaches to CSS Layout, Responsive Design, CSS Frameworks. Textbook 1: Ch. 4,5 RBT: L1, L2, L3			10
Module 3			
JavaScript: Client-Side Scripting, What is JavaScript and What can it do?, JavaScript Design Principles, Where does JavaScript Go?, Syntax, JavaScript Objects, The Document Object Model (DOM), JavaScript Events, Forms, Introduction to Server-Side Development with PHP, What is Server-Side Development, A Web Server's Responsibilities, Quick Tour of PHP, Program Control, Functions Textbook 1: Ch. 6, 8 RBT: L1, L2, L3			10
Module 4			
PHP Arrays and Superglobals, Arrays, \$_GET and \$_POST Superglobal Arrays, \$_SERVER Array, \$_FILES Array, Reading/Writing Files, PHP Classes and Objects, Object-Oriented Overview, Classes and Objects in PHP, Object Oriented Design, Error Handling and Validation, What are Errors and Exceptions?, PHP Error Reporting, PHP Error and Exception Handling Textbook 1: Ch. 9, 10 RBT: L1, L2, L3			10
Module 5			
Managing State, The Problem of State in Web Applications, Passing Information via Query Strings, Passing Information via the URL Path, Cookies, Serialization, Session State, HTML5 Web Storage, Caching, Advanced JavaScript and jQuery, JavaScript PseudoClasses, jQuery Foundations, AJAX, Asynchronous File Transmission, Animation, Backbone			10

MVC Frameworks, XML Processing and Web Services, XML Processing, JSON, Overview of Web Services. Textbook 1: Ch. 13, 15,17 RBT: L1, L2, L3	
Course Outcomes: The student will be able to :	
<ul style="list-style-type: none"> • Adapt HTML and CSS syntax and semantics to build web pages. • Construct and visually format tables and forms using HTML and CSS • Develop Client-Side Scripts using JavaScript and Server-Side Scripts using PHP to generate and display the contents dynamically. • Appraise the principles of object oriented development using PHP • Inspect JavaScript frameworks like jQuery and Backbone which facilitates developer to focus on core features. 	
Question Paper Pattern:	
<ul style="list-style-type: none"> • 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. 	
Textbooks:	
1. Randy Connolly, Ricardo Hoar, " Fundamentals of Web Development ", 1 st Edition, Pearson Education India. (ISBN:978-9332575271)	
Reference Books:	
<ol style="list-style-type: none"> 1. Robin Nixon, "Learning PHP, MySQL & JavaScript with jQuery, CSS and HTML5", 4thEdition, O'Reilly Publications, 2015. (ISBN:978-9352130153) 2. Luke Welling, Laura Thomson, "PHP and MySQL Web Development", 5th Edition, Pearson Education, 2016. (ISBN:978-9332582736) 3. Nicholas C Zakas, "Professional JavaScript for Web Developers", 3rd Edition, Wrox/Wiley India, 2012. (ISBN:978-8126535088) 4. David Sawyer Mcfarland, "JavaScript & jQuery: The Missing Manual", 1st Edition, O'Reilly/Shroff Publishers & Distributors Pvt Ltd, 2014 	
Mandatory Note:	
Distribution of CIE Marks is as follows (Total 40 Marks):	
<ul style="list-style-type: none"> • 20 Marks through IA Tests • 20 Marks through practical assessments 	
Maintain a copy of the report for verification during LIC visit.	
Possible list of practicals:	

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 “TEXTSHRINKING” 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, Programme, 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 case-insensitive comparison. [Note: Passing re.I as 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 Code	18CS641	CIE Marks	40
Number of Contact Hours/Week	3:0:0	SEE Marks	60
Total Number of Contact Hours	40	Exam Hours	03
CREDITS –3			
Course Learning Objectives: This course (18CS641) will enable students to:			
<ul style="list-style-type: none"> • 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			Contact 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
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
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
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
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
Course Outcomes: The student 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.

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.

Textbooks:

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.

OBJECT ORIENTED MODELING AND DESIGN
(Effective from the academic year 2018 -2019) SEMESTER
- VI

Course Code	18CS642	CIE Marks	40
Number of Contact Hours/Week	3:0:0	SEE Marks	60
Total Number of Contact Hours	40	Exam Hours	03
CREDITS –3			
Course Learning Objectives: This course (18CS642) will enable students to:			
<ul style="list-style-type: none"> • Describe the concepts involved in Object-Oriented modelling and their benefits. • Demonstrate concept of use-case model, sequence model and state chart model for a given problem. • Explain the facets of the unified process approach to design and build a Software system. • Translate the requirements into implementation for Object Oriented design. • Choose an appropriate design pattern to facilitate development procedure. 			
Module 1			Contact Hours
Advanced object and class concepts; Association ends; N-ary associations; Aggregation; Abstract classes; Multiple inheritance; Metadata; Reification; Constraints; Derived Data; Packages. State Modeling: Events, States, Transitions and Conditions, State Diagrams, State diagram behaviour. Text Book-1: 4, 5 RBT: L1, L2			08
Module 2			
UseCase Modelling and Detailed Requirements: Overview; Detailed object-oriented Requirements definitions; System Processes-A use case/Scenario view; Identifying Input and outputs-The System sequence diagram; Identifying Object Behaviour-The state chart Diagram; Integrated Object-oriented Models. Text Book-2:Chapter- 6:Page 210 to 250 RBT: L1, L2, L3			08
Module 3			
Process Overview, System Conception and Domain Analysis: Process Overview: Development stages; Development life Cycle; System Conception: Devising a system concept; elaborating a concept; preparing a problem statement. Domain Analysis: Overview of analysis; Domain Class model: Domain state model; Domain interaction model; Iterating the analysis. Text Book-1:Chapter- 10,11,and 12			08
Module 4			
Use case Realization :The Design Discipline within up iterations: Object Oriented DesignThe Bridge between Requirements and Implementation; Design Classes and Design within Class Diagrams; Interaction Diagrams-Realizing Use Case and defining methods; Designing with Communication Diagrams; Updating the Design Class Diagram; Package DiagramsStructuring the Major Components; Implementation Issues for Three-Layer Design. Text Book-2: Chapter 8: page 292 to 346 RBT: L1, L2, L3			08
Module 5			

Design Patterns: Introduction; what is a design pattern?, Describing design patterns, the catalogue of design patterns, Organizing the catalogue, How design patterns solve design problems, how to select a design patterns, how to use a design pattern; Creational patterns: prototype and singleton (only); structural patterns adaptor and proxy (only).	08
Text Book-3: Ch-1: 1.1, 1.3, 1.4, 1.5, 1.6, 1.7, 1.8,Ch-3,Ch-4. RBT: L1, L2, L3	
Course Outcomes: The student will be able to :	
<ul style="list-style-type: none"> Describe the concepts of object-oriented and basic class modelling. Draw class diagrams, sequence diagrams and interaction diagrams to solve problems. Choose and apply a befitting design pattern for the given problem. 	
Question Paper Pattern:	
<ul style="list-style-type: none"> 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. 	
Textbooks:	
<ol style="list-style-type: none"> Michael Blaha, James Rumbaugh: Object Oriented Modelling and Design with UML,2nd Edition, Pearson Education,2005 Satzinger, Jackson and Burd: Object-Oriented Analysis & Design with the Unified Process, Cengage Learning, 2005. Erich Gamma, Richard Helm, Ralph Johnson and john Vlissides: Design Patterns –Elements of Reusable Object-Oriented Software, Pearson Education,2007. 	
Reference Books:	
<ol style="list-style-type: none"> Grady Booch et. al.: Object-Oriented Analysis and Design with Applications,3rd Edition,Pearson Education,2007. 2.Frank Buschmann, RegineMeunier, Hans Rohnert, Peter Sommerlad, Michel Stal: Pattern – Oriented Software Architecture. A system of patterns , Volume 1, John Wiley and Sons.2007. 3. Booch, Jacobson, Rambaugh : Object-Oriented Analysis and Design with Applications, 3rd edition, pearson, Reprint 2013 	

CLOUD COMPUTING AND ITS APPLICATIONS (Effective from the academic year 2018 -2019) SEMESTER - VI			
Course Code	18CS643	CIE Marks	40
Number of Contact Hours/Week	3:0:0	SEE Marks	60
Total Number of Contact Hours	40	Exam Hours	03
CREDITS –3			
Course Learning Objectives: This course (18CS643) will enable students to:			
<ul style="list-style-type: none"> Explain the fundamentals of cloud computing Illustrate the cloud application programming and aneka platform Contrast different cloud platforms used in industry 			
Module 1			Contact Hours

<p>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 AppEngine, 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</p>	08
Module 2	
<p>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</p>	08
Module 3	
<p>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,</p>	08
<p>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</p>	
Module 4	
<p>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</p>	08

Module 5	
Cloud Platforms in Industry, Amazon Web Services, Compute Services, Storage Services, Communication Services, Additional Services, Google AppEngine, 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
Course Outcomes: The student will be able to :	
<ul style="list-style-type: none"> • 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. 	
Question Paper Pattern:	
<ul style="list-style-type: none"> • 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. 	
Textbooks:	
1. Rajkumar Buyya, Christian Vecchiola, and Thamarai Selvi Mastering Cloud. Computing McGraw Hill Education	
Reference Books:	
1. Dan C. Marinescu, Cloud Computing Theory and Practice, Morgan Kaufmann, Elsevier 2013.	

ADVANCED JAVA AND J2EE (Effective from the academic year 2018 -2019) SEMESTER - VI			
Course Code	18CS644	CIE Marks	40
Number of Contact Hours/Week	3:0:0	SEE Marks	60
Total Number of Contact Hours	40	Exam Hours	03
CREDITS –3			
Course Learning Objectives: This course (18CS644) will enable students to:			
<ul style="list-style-type: none"> • Identify the need for advanced Java concepts like Enumerations and Collections • Construct client-server applications using Java socket API • Make use of JDBC to access database through Java Programs • Adapt servlets to build server side programs • Demonstrate the use of JavaBeans to develop component-based Java software 			
Module 1		Contact Hours	

<p>Enumerations, Autoboxing and Annotations(metadata): Enumerations, Enumeration fundamentals, the values() and valueOf() Methods, java enumerations are class types, enumerations Inherits Enum, example, type wrappers, Autoboxing, Autoboxing and Methods, Autoboxing/Unboxing occurs in Expressions, Autoboxing/Unboxing, Boolean and character values, Autoboxing/Unboxing helps prevent errors, A word of Warning. Annotations, Annotation basics, specifying retention policy, Obtaining Annotations at run time by use of reflection, Annotated element Interface, Using Default values, Marker Annotations, Single Member annotations, Built-In annotations.</p> <p>Textbook 1: Lesson 12 RBT: L1, L2, L3</p>	08
Module 2	
<p>The collections and Framework: Collections Overview, Recent Changes to Collections, The Collection Interfaces, The Collection Classes, Accessing a collection Via an Iterator, Storing User Defined Classes in Collections, The Random Access Interface, Working With Maps, Comparators, The Collection Algorithms, Why Generic Collections?, The legacy Classes and Interfaces, Parting Thoughts on Collections.</p> <p>Text Book 1: Ch.17 RBT: L1, L2, L3</p>	08
Module 3	
<p>String Handling :The String Constructors, String Length, Special String Operations, String Literals, String Concatenation, String Concatenation with Other Data Types, String Conversion and toString() Character Extraction, charAt(), getChars(), getBytes() toCharArray(), String Comparison, equals() and equalsIgnoreCase(), regionMatches() startsWith() and endsWith(), equals() Versus == , compareTo() Searching Strings, Modifying a String, substring(), concat(), replace(), trim(), Data Conversion Using valueOf(), Changing the Case of Characters Within a String, Additional String Methods, StringBuffer , StringBuffer Constructors, length() and capacity(), ensureCapacity(), setLength(), charAt() and setCharAt(), getChars(),append(), insert(), reverse(), delete() and deleteCharAt(), replace(), substring(), Additional StringBuffer Methods, StringBuilder Text Book 1: Ch 15</p> <p>RBT: L1, L2, L3</p>	08
Module 4	
<p>Background; The Life Cycle of a Servlet; Using Tomcat for Servlet Development; A simple Servlet; The Servlet API; The Javax.servlet Package; Reading Servlet Parameter; The Javax.servlet.http package; Handling HTTP Requests and Responses; Using Cookies; Session Tracking. Java Server Pages (JSP): JSP, JSP Tags, Tomcat, Request String, User Sessions, Cookies, Session Objects</p> <p>Text Book 1: Ch 31 Text Book 2: Ch 11 RBT: L1, L2, L3</p>	08
Module 5	
<p>The Concept of JDBC; JDBC Driver Types; JDBC Packages; A Brief Overview of the JDBC process; Database Connection; Associating the JDBC/ODBC Bridge with the Database; Statement Objects; ResultSet; Transaction Processing; Metadata, Data types; Exceptions.</p> <p>Text Book 2: Ch 06 RBT: L1, L2, L3</p>	08
<p>Course Outcomes: The student will be able to :</p>	

<ul style="list-style-type: none"> • Interpret the need for advanced Java concepts like enumerations and collections in developing modular and efficient programs • Build client-server applications and TCP/IP socket programs • Illustrate database access and details for managing information using the JDBC API • Describe how servlets fit into Java-based web application architecture • Develop reusable software components using Java Beans
Question Paper Pattern:
<ul style="list-style-type: none"> • 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.
Textbooks:
<ol style="list-style-type: none"> 1. Herbert Schildt: JAVA the Complete Reference, 7th/9th Edition, Tata McGraw Hill, 2007. 2. Jim Keogh: J2EE-TheCompleteReference, McGraw Hill, 2007.
Reference Books:
<ol style="list-style-type: none"> 1. Y. Daniel Liang: Introduction to JAVA Programming, 7thEdition, Pearson Education, 2007. 2. Stephanie Bodoff et al: The J2EE Tutorial, 2nd Edition, Pearson Education,2004. 3. Uttam K Roy, Advanced JAVA programming, Oxford University press, 2015.

SYSTEM MODELLING AND SIMULATION			
(Effective from the academic year 2018 -2019) SEMESTER			
- VI			
Course Code	18CS645	CIE Marks	40
Number of Contact Hours/Week	3:0:0	SEE Marks	60
Total Number of Contact Hours	40	Exam Hours	03
CREDITS –3			
Course Learning Objectives: This course (18CS645) will enable students to:			
<ul style="list-style-type: none"> • 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 1			Contact 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
Module 2			

<p>Statistical Models in Simulation :Review of terminology and concepts, Useful statistical models,Discrete distributions. Continuous distributions,Poisson process, Empirical distributions.</p> <p>Queuing Models:Characteristics of queuing systems,Queuing notation,Long-run measures of performance of queuing systems,Long-run measures of performance of queuing systems cont...,Steady-state behavior of M/G/1 queue, Networks of queues,</p> <p>Textbook 1: Ch. 5,6.1 to 6.3, 6.4.1,6.6</p> <p>RBT: L1, L2, L3</p>	08
Module 3	
<p>Random-NumberGeneration:Properties of random numbers; Generation of pseudo-random numbers, Techniques for generating random numbers,Tests for Random Numbers,</p> <p>RandomVariate Generation: ,Inverse transform technique Acceptance-Rejection technique.</p> <p>Textbook 1: Ch. 7,8.1, 8.2</p> <p>RBT: L1, L2, L3</p>	08
Module 4	
<p>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.</p> <p>Estimation of Absolute Performance: Types of simulations with respect to output analysis ,Stochastic nature of output data, Measures of performance and their estimation,</p> <p>Contd.. Textbook 1: Ch. 9, 11.1 to 11.3</p> <p>RBT: L1, L2, L3</p>	08
Module 5	
<p>Measures of performance and their estimation,Output analysis for terminating simulations Continued...,Output analysis for steady-state simulations.</p> <p>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.</p>	08
<p>Textbook 1: Ch. 11.4, 11.5, 10 RBT:</p> <p>L1, L2, L3</p>	
<p>Course Outcomes: The student will be able to :</p> <ul style="list-style-type: none"> • 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. 	
<p>Question Paper Pattern:</p> <ul style="list-style-type: none"> • 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. 	
<p>Textbooks:</p> <p>1. Jerry Banks, John S. Carson II, Barry L. Nelson, David M. Nicol: Discrete-Event System Simulation, 5th Edition, Pearson Education, 2010.</p>	
<p>Reference Books:</p>	

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

MOBILE APPLICATION DEVELOPMENT
(OPEN ELECTIVE)
(Effective from the academic year 2018 -2019) SEMESTER
- VI

Course Code	18CS651	CIE Marks	40
Number of Contact Hours/Week	3:0:0	SEE Marks	60
Total Number of Contact Hours	40	Exam Hours	03
CREDITS –3			
Course Learning Objectives: This course (18CS651) will enable students to:			
<ul style="list-style-type: none"> • 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 			
Module – 1			Teaching Hours
Get started, Build your first app, Activities, Testing, debugging and using support libraries Textbook 1: Lesson 1,2,3 RBT: L1, L2			08
Module – 2			
User Interaction, Delightful user experience, Testing your UI Textbook 1: Lesson 4,5,6 RBT: L1, L2			08
Module – 3			
Background Tasks, Triggering, scheduling and optimizing background tasks Textbook 1: Lesson 7,8 RBT: L1, L2			08
Module – 4			
All about data, Preferences and Settings, Storing data using SQLite, Sharing data with providers, Loading data using Loaders Textbook 1: Lesson 9,10,11,12 RBT: L1, L2			08 content
Module – 5			
Permissions, Performance and Security, Firebase and AdMob, Publish// Textbook 1: Lesson 13,14,15 RBT: L1, L2			08
Course outcomes: The students should be able to:			
<ul style="list-style-type: none"> • Create, test and debug Android application by setting up Android development environment <input type="checkbox"/> • 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 			
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.

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

INTRODUCTION TO DATA STRUCTURES AND ALGORITHM
(OPEN ELECTIVE)
(Effective from the academic year 2018 -2019) SEMESTER
- VI

Course Code	18CS652	CIE Marks	40
Number of Contact Hours/Week	3:0:0	SEE Marks	60
Total Number of Contact Hours	40	Exam Hours	03
CREDITS –3			
Course Learning Objectives: This course (18CS652) will enable students to:			
<ul style="list-style-type: none"> • Identify different data structures in C programming language □ Appraise the use of data structures in problem solving • Implement data structures using C programming language. 			
Module 1			Contact Hours
Introduction to C, constants, variables, data types, input output operations, operators and expressions, control statements, arrays, strings, built-in functions, user defined functions, structures, unions and pointers Text Book 1: Chapter 1 and 2 RBT: L1, L2			08
Module 2			
Algorithms, Asymptotic notations, Introduction to data structures, Types of data structures, Arrays. Text Book 1: Chapter 3 and 4 RBT: L1, L2			08
Module 3			
Linked lists, Stacks Text Book 1: Chapter 5 and 6 RBT: L1, L2			08
Module 4			
Queues, Trees Text Book 1: Chapter 7 and 8 RBT: L1, L2			08
Module 5			
Graphs, Sorting ,(selection, insertion, bubble, quick)and searching(Linear, Binary, Hash) Text Book 1: Chapter 7 and 8 RBT: L1, L2			08
Course Outcomes: The student will be able to :			
<ul style="list-style-type: none"> • Identify different data structures in C programming language □ Appraise the use of data structures in problem solving • Implement data structures using C programming language. 			
Question Paper Pattern:			
<ul style="list-style-type: none"> • 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. 			
Textbooks:			

1. Data structures using C , E Balagurusamy, McGraw Hill education (India) Pvt. Ltd, 2013.

Reference Books:

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

PROGRAMMING IN JAVA
(OPEN ELECTIVE)
(Effective from the academic year 2018 -2019) SEMESTER
- VI

Course Code	18CS653	CIE Marks	40
Number of Contact Hours/Week	3:0:0	SEE Marks	60
Total Number of Contact Hours	40	Exam Hours	03
CREDITS –3			
Course Learning Objectives: This course (18CS653) will enable students to:			
<ul style="list-style-type: none"> • 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 – 1			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 Text book 1: Ch 2, Ch 3 RBT: L1, L2			08
Module – 2			
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. Text book 1: Ch 4, Ch 5 RBT: L1, L2			08
Module – 3			
Introducing Classes: Class Fundamentals, Declaring Objects, Assigning Object Reference Variables, Introducing Methods, Constructors, The this Keyword, Garbage Collection, The finalize() Method, A Stack Class, 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. Text book 1: Ch 6, Ch 7.1-7.9, Ch 8. RBT: L1, L2			08
Module – 4			

<p>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.</p>	08
<p>Text book 1: Ch 9, Ch 10 RBT: L1, L2</p>	
<p>Module – 5</p>	
<p>Enumerations, Type Wrappers, I/O, Applets, and Other Topics: I/O Basics, Reading Console Input, Writing Console Output, The PrintWriter Class, Reading and Writing Files, Applet Fundamentals, The transient and volatile Modifiers, Using instanceof, strictfp, Native Methods, Using assert, Static Import, Invoking Overloaded Constructors Through this(), String 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.</p> <p>Text book 1: Ch 12.1,12.2, Ch 13, Ch 15 RBT: L1, L2</p>	08
<p>Course outcomes: The students should be able to:</p> <ul style="list-style-type: none"> • Explain the object-oriented concepts and JAVA. • Develop computer programs to solve real world problems in Java. <p>Develop simple GUI interfaces for a computer program to interact with users</p>	
<p>Question Paper Pattern:</p> <ul style="list-style-type: none"> • 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. 	
<p>Text Books:</p> <p>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)</p>	
<p>Reference Books:</p> <p>1. Cay S Horstmann, "Core Java - Vol. 1 Fundamentals", Pearson Education, 10th Edition, 2016. 2. Raoul-Gabriel Urma, Mario Fusco, Alan Mycroft, "Java 8 in Action", Dreamtech Press/Manning Press, 1st Edition, 2014.</p>	

<p>INTRODUCTION TO OPERATING SYSTEM (OPEN ELECTIVE) (Effective from the academic year 2018 -2019) SEMESTER – VII</p>			
Course Code	18CS654	CIE Marks	40
Number of Contact Hours/Week	3:0:0	SEE Marks	60
Total Number of Contact Hours	40	Exam Hours	03
<p>CREDITS –3</p>			

Course Learning Objectives: This course (18CS654) will enable students to:	
<ul style="list-style-type: none"> • Explain the fundamentals of operating system • Comprehend multithreaded programming, process management, memory management and storage management. • Familiar with various types of operating systems 	
Module – 1	Teaching Hours
<p>Introduction: What OS do, Computer system organization, architecture, structure, Operations, Process, memory and storage management, Protection and security, Distributed systems, Special purpose systems, computing environments.</p> <p>System Structure: OS Services, User OSI, System calls, Types of system calls, System programs, OS design and implementation, OS structure, Virtual machines, OS generation, system boot</p> <p>Textbook1: Chapter 1, 2 RBT: L1, L2</p>	08
Module – 2	
<p>Process Concept: Overview, Process scheduling, Operations on process, IPC, Examples in IPC, Communication in client-server systems.</p> <p>Multithreaded Programming: Overview, Models, Libraries, Issues, OS Examples</p> <p>Textbook1: Chapter 3,4 RBT: L1, L2</p>	08
Module – 3	
<p>Process Scheduling: Basic concept, Scheduling criteria, Algorithm, multiple processor scheduling, thread scheduling, OS Examples, Algorithm Evaluation.</p> <p>Synchronization: Background, the critical section problem, Petersons solution, Synchronization hardware, Semaphores, Classic problems of synchronization, Monitors, Synchronization examples, Atomic transactions</p> <p>Textbook1: Chapter 5, 6 RBT: L1, L2</p>	08
Module – 4	
<p>Deadlocks: System model, Deadlock characterization, Method of handling deadlock, Deadlock prevention, Avoidance, Detection, Recovery from deadlock</p> <p>Memory management strategies: Background, swapping, contiguous memory allocation, paging, structure of page table, segmentation,</p> <p>Textbook1: Chapter 7, 8 RBT: L1, L2</p>	08
Module – 5	
<p>Virtual Memory management: Background, Demand paging, Copy-on-write, Page replacement, allocation of frames, Trashing, Memory mapped files, Allocating Kernel memory, Operating system examples</p>	08
<p>File system: File concept, Access methods, Directory structure, File system mounting, File sharing, protection</p> <p>Textbook1: Chapter 9, 10 RBT: L1, L2</p>	
Course outcomes: The students should be able to:	
<ul style="list-style-type: none"> • Explain the fundamentals of operating system • Comprehend process management, memory management and storage management. • Familiar with various types of operating systems 	

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 Books:

1. A. Silberschatz, P B Galvin, G Gagne, Operating systems, 7th edition, John Wiley and sons,.

Reference Books:

1. William Stalling,"Operating Systems: Internals and Design Principles", Pearson Education, 1st Edition, 2018.
2. Andrew S Tanenbaum, Herbert BOS, "Modern Operating Systems", Pearson Education, 4th Edition, 2016

SYSTEM SOFTWARE LABORATORY
(Effective from the academic year 2018 -2019) SEMESTER
- VI

Course Code	18CSL66	CIE Marks	40
Number of Contact Hours/Week	0:2:2	SEE Marks	60
Total Number of Lab Contact Hours	36	Exam Hours	03

Credits – 2

Course Learning Objectives: This course (18CSL66) will enable students to:

- 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

Descriptions (if any):

Exercises to be prepared with minimum three files (Where ever necessary):

1. Header file.
2. Implementation file.
3. Application file where main function will be present.

The idea behind using three files is to differentiate between the developer and user sides. In the developer side, all the three files could be made visible. For the user side only header file and application files could be made visible, which means that the object code of the implementation file could be given to the user along with the interface given in the header file, hiding the source file, if required. Avoid I/O operations (printf/scanf) and use *data input file* where ever it is possible.

Programs List:

Installation procedure of the required software must be demonstrated, carried out in groups and documented in the journal.

1.	
a.	Write a LEX program to recognize valid <i>arithmetic expression</i> . Identifiers in the expression could be only integers and operators could be + and *. Count the identifiers & operators present and print them separately.
b.	Write YACC program to evaluate <i>arithmetic expression</i> involving operators: +, -, *, and /
2.	Develop, Implement and Execute a program using YACC tool to recognize all strings ending with <i>b</i> preceded by <i>n a's</i> using the grammar <i>aⁿ b</i> (note: input <i>n</i> value)
3.	Design, develop and implement YACC/C program to construct <i>Predictive / LL(1) Parsing Table</i> for the grammar rules: <i>A → aBa</i> , <i>B → bB</i> ϵ . Use this table to parse the sentence: <i>abba\$</i>
4.	Design, develop and implement YACC/C program to demonstrate <i>Shift Reduce Parsing</i> technique for the grammar rules: <i>E → E+T</i> <i>T, T → T*F</i> <i>F, F → (E)</i> <i>id</i> and parse the sentence: <i>id + id * id</i> .

5.	Design, develop and implement a C/Java program to generate the machine code using <i>Triples</i> for the statement $A = -B * (C + D)$ whose intermediate code in three-address form: $T1 = -B$ $T2 = C + D$ $T3 = T1 + T2$ $A = T3$
6.	
a.	Write a LEX program to eliminate <i>comment lines</i> in a C program and copy the resulting program into a separate file.
b.	Write YACC program to recognize valid <i>identifier, operators and keywords</i> in the given text (C program) file.
7.	Design, develop and implement a C/C++/Java program to simulate the working of Shortest remaining time and Round Robin (RR) scheduling algorithms. Experiment with different quantum sizes for RR algorithm.
8.	Design, develop and implement a C/C++/Java program to implement Banker's algorithm. Assume suitable input required to demonstrate the results
9.	Design, develop and implement a C/C++/Java program to implement page replacement algorithms LRU and FIFO. Assume suitable input required to demonstrate the results.
Laboratory Outcomes: The student should be able to:	
<ul style="list-style-type: none"> • 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:	
<ul style="list-style-type: none"> • Experiment distribution <ul style="list-style-type: none"> ○ 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 (<i>Courseed to change in accordance with university regulations</i>) <ul style="list-style-type: none"> m) For laboratories having only one part – Procedure + Execution + Viva-Voce: 15+70+15 = 100 Marks n) For laboratories having PART A and PART B <ul style="list-style-type: none"> i. Part A – Procedure + Execution + Viva = 6 + 28 + 6 = 40 Marks ii. Part B – Procedure + Execution + Viva = 9 + 42 + 9 = 60 Marks 	

COMPUTER GRAPHICS LABORATORY WITH MINI PROJECT
(Effective from the academic year 2018 -2019) SEMESTER
- VI

Course Code	18CSL67	CIE Marks	40
Number of Contact Hours/Week	0:2:2	SEE Marks	60
Total Number of Lab Contact Hours	36	Exam Hours	03

Credits – 2	
Course Learning Objectives: This course (18CSL67) will enable students to:	
<ul style="list-style-type: none"> • Demonstrate simple algorithms using OpenGL Graphics Primitives and attributes. • Implementation of line drawing and clipping algorithms using OpenGL functions • Design and implementation of algorithms Geometric transformations on both 2D and 3D objects. 	
Descriptions (if any): --	
Installation procedure of the required software must be demonstrated, carried out in groups and documented in the journal.	
Programs List:	
PART A	
Design, develop, and implement the following programs using OpenGL API	
1.	Implement Brenham’s line drawing algorithm for all types of slope. Refer:Text-1: Chapter 3.5 Refer:Text-2: Chapter 8
2.	Create and rotate a triangle about the origin and a fixed point. Refer:Text-1: Chapter 5-4
3.	Draw a colour cube and spin it using OpenGL transformation matrices. Refer:Text-2: Modelling a Coloured Cube
4.	Draw a color cube and allow the user to move the camera suitably to experiment with perspective viewing. Refer:Text-2: Topic: Positioning of Camera
5.	Clip a lines using Cohen-Sutherland algorithm Refer:Text-1: Chapter 6.7 Refer:Text-2: Chapter 8
6.	To draw a simple shaded scene consisting of a tea pot on a table. Define suitably the position and properties of the light source along with the properties of the surfaces of the solid object used in the scene. Refer:Text-2: Topic: Lighting and Shading
7.	Design, develop and implement recursively subdivide a tetrahedron to form 3D sierpinski gasket. The number of recursive steps is to be specified by the user. Refer: Text-2: Topic: sierpinski gasket.
8.	Develop a menu driven program to animate a flag using Bezier Curve algorithm Refer: Text-1: Chapter 8-10
9.	Develop a menu driven program to fill the polygon using scan line algorithm
PART B MINI PROJECT	
Student should develop mini project on the topics mentioned below or similar applications using Open GL API. Consider all types of attributes like color, thickness, styles, font, background, speed etc., while doing mini project. (During the practical exam: the students should demonstrate and answer Viva-Voce) Sample Topics: Simulation of concepts of OS, Data structures, algorithms etc.	
Laboratory Outcomes: The student should be able to:	
<input type="checkbox"/> Apply the concepts of computer graphics	
<ul style="list-style-type: none"> • Implement computer graphics applications using OpenGL • Animate real world problems using OpenGL 	
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 accordance with university regulations*)
 - o) For laboratories having only one part – Procedure + Execution + Viva-Voce: $15+70+15 = 100$ Marks
 - p) For laboratories having PART A and PART B
 - i. Part A – Procedure + Execution + Viva = $6 + 28 + 6 = 40$ Marks
 - ii. Part B – Procedure + Execution + Viva = $9 + 42 + 9 = 60$ Marks

MOBILE APPLICATION DEVELOPMENT
(Effective from the academic year 2018 -2019) SEMESTER
- VI

Course Code	18CSMP68	IA Marks	40
Number of Contact Hours/Week	0:0:2	Exam Marks	60
Total Number of Contact Hours	3 Hours/Week	Exam Hours	03

CREDITS – 02

Laboratory Objectives: This laboratory (18CSMP68) will enable students to


- Learn and acquire the art of Android Programming.
- Configure Android studio to run the applications.
- Understand and implement Android's User interface functions.
- Create, modify and query on SQLite database.
- Inspect different methods of sharing data using services.

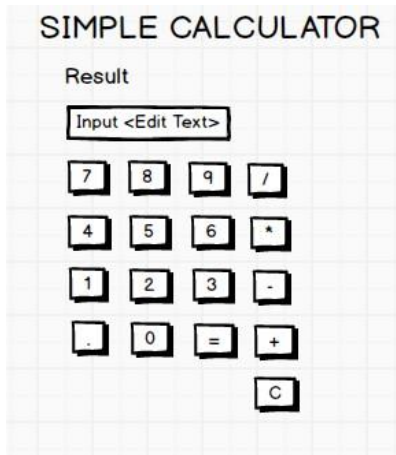
Descriptions (if any):

1. The installation procedure of the Android Studio/Java software must be demonstrated and carried out in groups.
2. Students should use the latest version of Android Studio/Java/ Kotlin to execute these programs. Diagrams given are for representational purposes only, students are expected to improvise on them.
3. **Part B programs should be developed as an application and are to be demonstrated as a mini project in a group by adding extra features or the students can also develop their application and demonstrate it as a mini-project. (Projects/programs are not limited to the list given in Part B).**

Programs List:

PART – A

1	<p>Create an application to design a Visiting Card. The Visiting card should have a company logo at the top right corner. The company name should be displayed in Capital letters, aligned to the center. Information like the name of the employee, job title, phone number, address, email, fax and the website address is to be displayed. Insert a horizontal line between the job title and the phone number.</p> <div style="text-align: center;">  </div>
2	<p>Develop an Android application using controls like Button, TextView, EditText for designing a calculator having basic functionality like Addition, Subtraction, Multiplication, and Division.</p>



3

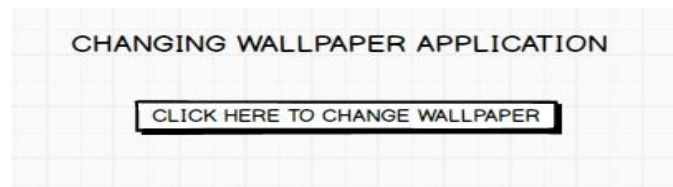
Create a SIGN Up activity with Username and Password. Validation of password should happen based on the following rules:

- Password should contain uppercase and lowercase letters.
- Password should contain letters and numbers.
- Password should contain special characters.
- Minimum length of the password (the default value is 8).

On successful **SIGN UP** proceed to the next Login activity. Here the user should **SIGN IN** using the Username and Password created during signup activity. If the Username and Password are matched then navigate to the next activity which displays a message saying “Successful Login” or else display a toast message saying “Login Failed”. The user is given only two attempts and after that display a toast message saying “Failed Login Attempts” and disable the SIGN IN button. Use Bundle to transfer information from one activity to another.

SIGNUP ACTIVITY	LOGIN ACTIVITY
Username: <input type="text"/>	Username: <input type="text"/>
Password: <input type="password"/>	Password: <input type="password"/>
<input type="button" value="SIGN UP"/>	<input type="button" value="SIGN IN"/>

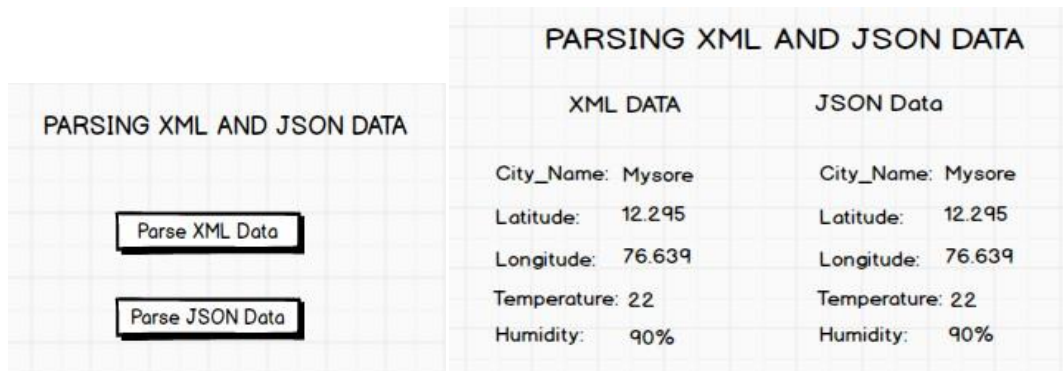
- 4 Develop an application to set an image as wallpaper. On click of a button, the wallpaper image should start to change randomly every 30 seconds.




- 5 Write a program to create an activity with two buttons START and STOP. On pressing of the START button, the activity must start the counter by displaying the numbers from One and the counter must keep on counting until the STOP button is pressed. Display the counter value in a TextView control.



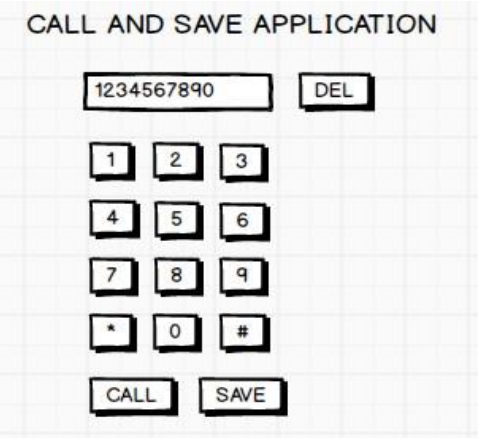
- 6 Create two files of XML and JSON type with values for City_Name, Latitude, Longitude, Temperature, and Humidity. Develop an application to create an activity with two buttons to parse the XML and JSON files which when clicked should display the data in their respective layouts side by side.



7 Develop a simple application with one EditText so that the user can write some text in it. Create a button called “Convert Text to Speech” that converts the user input text into voice.




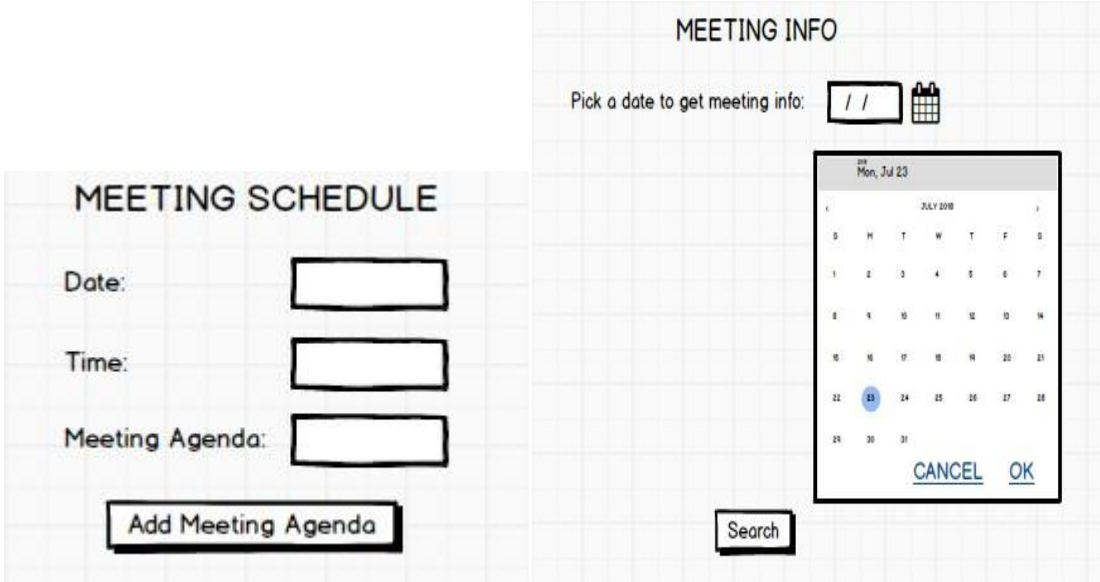
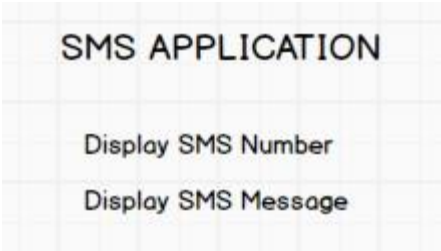
8 Create an activity like a phone dialer with CALL and SAVE buttons. On pressing the CALL button, it must call the phone number and on pressing the SAVE button it must save the number to the phone contacts.

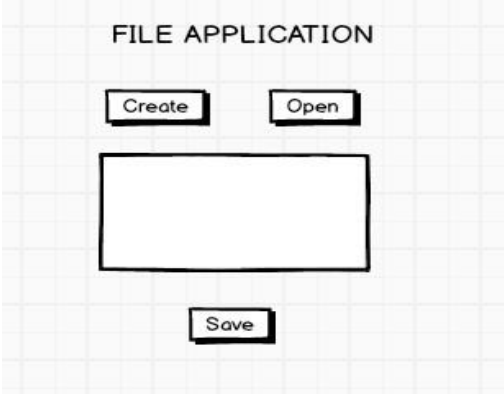
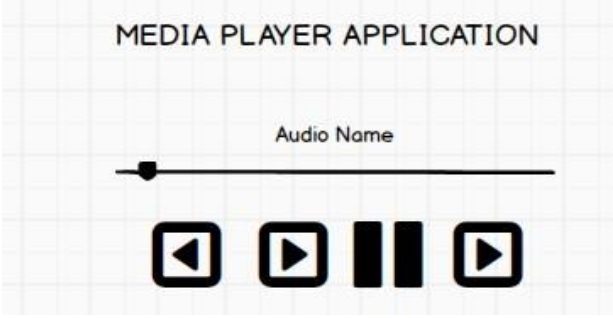
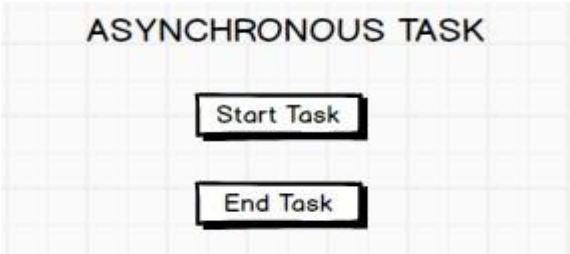


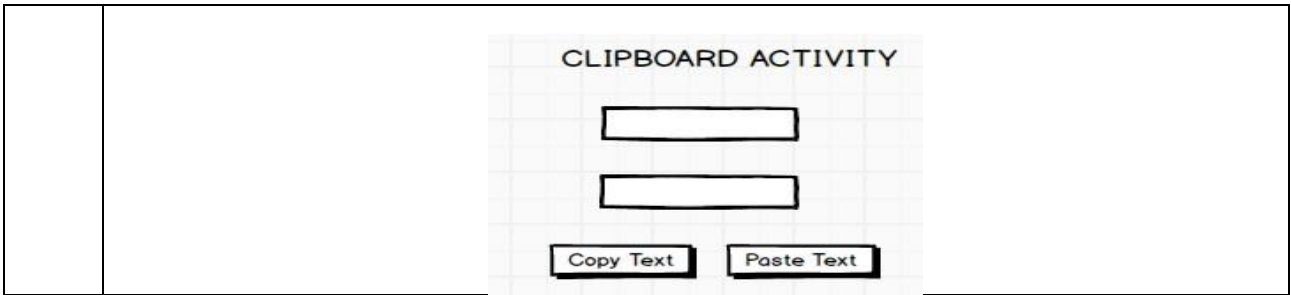
PART - B

1 Write a program to enter Medicine Name, Date and Time of the Day as input from the user and store it in the SQLite database. Input for Time of the Day should be either Morning or Afternoon or Evening or Night. Trigger an alarm based on the Date and Time of the Day and display the Medicine Name.



<p>2</p>	<p>Develop a content provider application with an activity called “Meeting Schedule” which takes Date, Time and Meeting Agenda as input from the user and store this information into the SQLite database. Create another application with an activity called “Meeting Info” having DatePicker control, which on the selection of a date should display the Meeting Agenda information for that particular date, else it should display a toast message saying “No Meeting on this Date”.</p> 
<p>3</p>	<p>Create an application to receive an incoming SMS which is notified to the user. On clicking this SMS notification, the message content and the number should be displayed on the screen. Use appropriate emulator control to send the SMS message to your application.</p> 
<p>4</p>	<p>Write a program to create an activity having a Text box, and also Save, Open and Create buttons. The user has to write some text in the Text box. On pressing the Create button the text should be saved as a text file in Mksdcard. On subsequent changes to the text, the Save button should be pressed to store the latest content to the same file. On pressing the Open button, it should display the contents from the previously stored files in the Text box. If the user tries to save the contents in the Textbox to a file without creating it, then a toast message has to be displayed saying “First Create a File”.</p>

	
5	<p>Create an application to demonstrate a basic media player that allows the user to Forward, Backward, Play and Pause an audio. Also, make use of the indicator in the seek bar to move the audio forward or backward as required.</p> 
6	<p>Develop an application to demonstrate the use of Asynchronous tasks in android. The asynchronous task should implement the functionality of a simple moving banner. On pressing the Start Task button, the banner message should scroll from right to left. On pressing the Stop Task button, the banner message should stop. Let the banner message be “Demonstration of Asynchronous Task”.</p> 
7	<p>Develop an application that makes use of the clipboard framework for copying and pasting of the text. The activity consists of two EditText controls and two Buttons to trigger the copy and paste functionality.</p>



8 Create an AIDL service that calculates Car Loan EMI. The formula to calculate EMI is

$$E = P * (r(1+r)^n)/((1+r)^n-1) \text{ where}$$

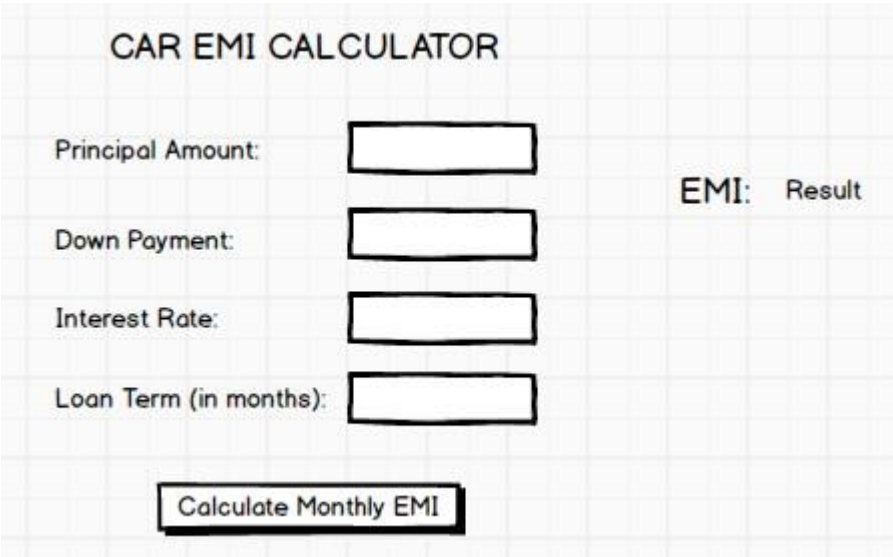
E = The EMI payable on the car loan amount

P = The Car loan Principal Amount

r = The interest rate value computed on a monthly basis

n = The loan tenure in the form of months

The down payment amount has to be deducted from the principal amount paid towards buying the Car. Develop an application that makes use of this AIDL service to calculate the EMI. This application should have four EditText to read the PrincipalAmount, Down Payment, Interest Rate, Loan Term (in months) and a button named as “Calculate Monthly EMI”. On click of this button, the result should be shown in a TextView. Also, calculate the EMI by varying the Loan Term and Interest Rate values.



Laboratory Outcomes: After studying these laboratory programs, students 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.

□ Infer the role of permissions and security for Android applications.

Procedure to Conduct Practical Examination	
<ul style="list-style-type: none"> • Experiment distribution <ul style="list-style-type: none"> ○ 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 accordance with university regulations) <ul style="list-style-type: none"> ○ For laboratories having only one part – Procedure + Execution + Viva-Voce: 15+70+15= 100 Marks <ul style="list-style-type: none"> ○ For laboratories having PART A and PART B <ul style="list-style-type: none"> i. Part A – Procedure + Execution + Viva = 6 + 28 + 6 = 40 Marks ii. Part B – Procedure + Execution + Viva = 9 + 42 + 9 = 60 Marks 	
Text Books:	
1. Google Developer Training, " Android Developer Fundamentals Course – Concept Reference ", Google Developer Training Team, 2017. https://www.gitbook.com/book/google-developer-training/android-developer-fundamentalscourse-concepts/details (Download pdf file from the above link)	
Reference Books:	
1. Erik Hellman, " Android Programming – Pushing the Limits ", 1 st Edition, Wiley India Pvt Ltd, 2014. ISBN-13: 978-8126547197 2. Dawn Griffiths and David Griffiths, " Head First Android Development ", 1 st Edition, O'Reilly SPD Publishers, 2015. ISBN-13: 978-9352131341 3. Bill Phillips, Chris Stewart and Kristin Marsicano, " Android Programming: The Big Nerd Ranch Guide ", 3 rd Edition, Big Nerd Ranch Guides, 2017. ISBN-13: 978-0134706054	

ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING (Effective from the academic year 2018 -2019) SEMESTER – VII			
Course Code	18CS71	CIE Marks	40
Number of Contact Hours/Week	4:0:0	SEE Marks	60
Total Number of Contact Hours	50	Exam Hours	03
CREDITS –4			
Course Learning Objectives: This course (18CS71) will enable students to:			
<ul style="list-style-type: none"> • Explain Artificial Intelligence and Machine Learning • Illustrate AI and ML algorithm and their use in appropriate applications 			
Module 1			Contact Hours

What is artificial intelligence?, Problems, problem spaces and search, Heuristic search techniques Textbook 1: Chapter 1, 2 and 3 RBT: L1, L2	10
Module 2	
Knowledge representation issues, Predicate logic, Representaiton knowledge using rules. Concpet Learning: Concept learning task, Concpet learning as search, Find-S algorithm, Candidate Elimination Algorithm, Inductive bias of Candidate Elimination Algorithm. Textbook 1: Chapter 4, 5 and 6 Textbook2: Chapter 2 (2.1-2.5, 2.7) RBT: L1, L2, L3	10
Module 3	
Decision Tree Learning: Introduction, Decision tree representation, Appropriate problems, ID3 algorithm. Aritificil Nueral Network: Introduction, NN representation, Appropriate problems, Perceptrons, Backpropagation algorithm. Textbook2: Chapter 3 (3.1-3.4), Chapter 4 (4.1-4.5) RBT: L1, L2, L3	10
Module 4	
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 Textbook2: Chapter 6 RBT: L1, L2, L3	10
Module 5	
Instance-Base Learning: Introduction, k-Nearest Neighbour Learning, Locally weighted regression, Radial basis function, Case-Based reasoning. Reinforcement Learning: Introduction, The learning task, Q-Learning. Textbook 1: Chapter 8 (8.1-8.5), Chapter 13 (13.1 – 13.3) RBT: L1, L2, L3	10
Course Outcomes: The student will be able to :	
<ul style="list-style-type: none"> • Appaise the theory of Artificial intelligence and Machine Learning. • Illustrate the working of AI and ML Algorithms. • Demonstrate the applications of AI and ML. 	
Question Paper Pattern:	
<ul style="list-style-type: none"> • The question paper will have ten questions. • Each full Question consisting of 20 marks 	
<ul style="list-style-type: none"> • 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. 	
Textbooks:	
<ol style="list-style-type: none"> 1. Tom M Mitchell, “Machine Lerning”, 1st Edition, McGraw Hill Education, 2017. 2. Elaine Rich, Kevin K and S B Nair, “Artificial Intelligence”, 3rd Edition, McGraw Hill Education, 2017. 	
Reference Books:	

1. Saroj Kaushik, Artificial Intelligence, Cengage learning
2. Stuart Russell, Peter Norving , Artificial Intelligence: A Modern Approach, Pearson Education 2nd Edition
3. Aurélien Geron, "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, h The Elements of Statistical Learning, 2nd edition, springer series in statistics.
5. Ethem Alpaydm, Introduction to machine learning, second edition, MIT press
6. Srinivasa K G and Shreedhar, “ Artificial Intelligence and Machine Learning”, Cengage

BIG DATA AND ANALYTICS
(Effective from the academic year 2018 -2019) SEMESTER
– VII

Course Code	18CS72	CIE Marks	40
Number of Contact Hours/Week	4:0:0	SEE Marks	60
Total Number of Contact Hours	50	Exam Hours	03
CREDITS –4			
Course Learning Objectives: This course (18CS72) will enable students to:			
<ul style="list-style-type: none"> • Understand fundamentals of Big Data analytics • Explore the Hadoop framework and Hadoop Distributed File system • Illustrate the concepts of NoSQL using MongoDB and Cassandra for Big Data • Employ MapReduce programming model to process the big data • Understand various machine learning algorithms for Big Data Analytics, Web Mining and Social Network Analysis. 			
Module 1			Contact Hours
Introduction to Big Data Analytics: Big Data, Scalability and Parallel Processing, Designing Data Architecture, Data Sources, Quality, Pre-Processing and Storing, Data Storage and Analysis, Big Data Analytics Applications and Case Studies. Text book 1: Chapter 1: 1.2 -1.7 RBT: L1, L2, L3			10
Module 2			
Introduction to Hadoop (T1): Introduction, Hadoop and its Ecosystem, Hadoop Distributed File System, MapReduce Framework and Programming Model, Hadoop Yarn, Hadoop Ecosystem Tools. Hadoop Distributed File System Basics (T2): HDFS Design Features, Components, HDFS User Commands. Essential Hadoop Tools (T2): Using Apache Pig, Hive, Sqoop, Flume, Oozie, HBase. Text book 1: Chapter 2 :2.1-2.6 Text Book 2: Chapter 3 Text Book 2: Chapter 7 (except walk throughs) RBT: L1, L2, L3			10
Module 3			
NoSQL Big Data Management, MongoDB and Cassandra: Introduction, NoSQL Data Store, NoSQL Data Architecture Patterns, NoSQL to Manage Big Data, Shared-Nothing Architecture for Big Data Tasks, MongoDB, Databases, Cassandra Databases. Text book 1: Chapter 3: 3.1-3.7 RBT: L1, L2, L3			10
Module 4			
MapReduce, Hive and Pig: Introduction, MapReduce Map Tasks, Reduce Tasks and MapReduce Execution, Composing MapReduce for Calculations and Algorithms, Hive, HiveQL, Pig. Text book 1: Chapter 4: 4.1-4.6 RBT: L1, L2, L3			10

Module 5		
<p>Machine Learning Algorithms for Big Data Analytics: Introduction, Estimating the relationships, Outliers, Variances, Probability Distributions, and Correlations, Regression analysis, Finding Similar Items, Similarity of Sets and Collaborative Filtering, Frequent Itemsets and Association Rule Mining.</p> <p>Text, Web Content, Link, and Social Network Analytics: Introduction, Text mining, Web Mining, Web Content and Web Usage Analytics, Page Rank, Structure of Web and analyzing a Web Graph, Social Network as Graphs and Social Network Analytics:</p> <p>Text book 1: Chapter 6: 6.1 to 6.5 Text book 1: Chapter 9: 9.1 to 9.5</p>		10
Course Outcomes: The student will be able to:		
<ul style="list-style-type: none"> • Understand fundamentals of Big Data analytics. • Investigate Hadoop framework and Hadoop Distributed File system. • Illustrate the concepts of NoSQL using MongoDB and Cassandra for Big Data. • Demonstrate the MapReduce programming model to process the big data along with Hadoop tools. • Use Machine Learning algorithms for real world big data. • Analyze web contents and Social Networks to provide analytics with relevant visualization tools. 		
Question Paper Pattern:		
<ul style="list-style-type: none"> • 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. 		
Textbooks:		
<ol style="list-style-type: none"> 1. Raj Kamal and Preeti Saxena, “Big Data Analytics Introduction to Hadoop, Spark, and Machine-Learning”, McGraw Hill Education, 2018 ISBN: 9789353164966, 9353164966 2. Douglas Eadline, "Hadoop 2 Quick-Start Guide: Learn the Essentials of Big Data Computing in the Apache Hadoop 2 Ecosystem", 1stEdition, Pearson Education, 2016. ISBN13: 978-9332570351 		
Reference Books:		
<ol style="list-style-type: none"> 1. Tom White, “Hadoop: The Definitive Guide”, 4th Edition, O’Reilly Media, 2015.ISBN-13: 9789352130672 2. Boris Lublinsky, Kevin T Smith, Alexey Yakubovich, "Professional Hadoop Solutions", 1stEdition, Wrox Press, 2014ISBN-13: 978-8126551071 3. Eric Sammer, "Hadoop Operations: A Guide for Developers and Administrators", 1stEdition, O’Reilly Media, 2012.ISBN-13: 978-9350239261 4. Arshdeep Bahga, Vijay Madiseti, "Big Data Analytics: A Hands-On Approach", 1st Edition, VPT Publications, 2018. ISBN-13: 978-0996025577 		

SOFTWARE ARCHITECTURE AND DESIGN PATTERNS
(Effective from the academic year 2018 -2019) SEMESTER
- VII

Course Code	18CS731	CIE Marks	40
Number of Contact Hours/Week	3:0:0	SEE Marks	60

Total Number of Contact Hours	40	Exam Hours	03
CREDITS –3			
Course Learning Objectives: This course (18CS731) will enable students to:			
<ul style="list-style-type: none"> • Learn How to add functionality to designs while minimizing complexity. • What code qualities are required to maintain to keep code flexible? • To Understand the common design patterns. • To explore the appropriate patterns for design problems 			
Module 1			Contact Hours
Introduction: what is a design pattern? describing design patterns, the catalog of design pattern, organizing the catalog, how design patterns solve design problems, how to select a design pattern, how to use a design pattern. A Notation for Describing Object-Oriented Systems Textbook 1: Chapter 1 and 2.7 Analysis a System: overview of the analysis phase, stage 1: gathering the requirements functional requirements specification, defining conceptual classes and relationships, using the knowledge of the domain. Design and Implementation, discussions and further reading. Textbook 1: Chapter 6 RBT: L1, L2, L3			08
Module 2			
Design Pattern Catalog: Structural patterns, Adapter, bridge, composite, decorator, facade, flyweight, proxy. Textbook 2: chapter 4 RBT: L1, L2, L3			08
Module 3			
BehavioralPatterns: Chain of Responsibility, Command, Interpreter, Iterator, Mediator, Memento, Observer, State, Template Method Textbook 2: chapter 5 RBT: L1, L2, L3			08
Module 4			
Interactive systems and the MVC architecture: Introduction, The MVC architectural pattern, analyzing a simple drawing program, designing the system, designing of the subsystems, getting into implementation, implementing undo operation, drawing incompleteitems, adding a new feature, pattern-based solutions. Textbook 1: Chapter 11 RBT: L1, L2, L3			08
Module 5			
Designing with Distributed Objects: Client server system, java remote method invocation, implementing an object-oriented system on the web (discussions and further reading) a note on input and output, selection statements, loops arrays. Textbook 1: Chapter 12 RBT: L1, L2, L3			08
Course Outcomes: The student will be able to :			
<ul style="list-style-type: none"> • Design and implement codes with higher performance and lower complexity • Be aware of code qualities needed to keep code flexible 			

- Experience core design principles and be able to assess the quality of a design with respect to these principles.
- Capable of applying these principles in the design of object oriented systems.
- Demonstrate an understanding of a range of design patterns. Be capable of comprehending a design presented using this vocabulary.
- Be able to select and apply suitable patterns in specific contexts

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.

Textbooks:

1. Brahma Dathan, Sarnath Rammath, Object-oriented analysis, design and implementation, Universities Press,2013
2. Erich Gamma, Richard Helan, Ralph Johman, John Vlissides , Design Patterns, Pearson Publication,2013.

Reference Books:

1. Frank Bachmann, RegineMeunier, Hans Rohnert “Pattern Oriented Software Architecture” –Volume 1, 1996.
2. William J Brown et al., "Anti-Patterns: Refactoring Software, Architectures and Projects in Crisis", John Wiley, 1998.

HIGH PERFORMANCE COMPUTING (Effective from the academic year 2018 -2019) SEMESTER - VII			
Course Code	18CS732	CIE Marks	40
Number of Contact Hours/Week	3:0:0	SEE Marks	60
Total Number of Contact Hours	40	Exam Hours	03
CREDITS –3			
Course Learning Objectives: This course (18CS732) will enable students to:			
<ul style="list-style-type: none"> • Introduce students the design, analysis, and implementation, of high performance computational science and engineering applications. • Illustrate on advanced computer architectures, parallel algorithms, parallel languages, and performance-oriented computing. 			
Module – 1			Contact Hours

<p>Introduction to Parallel Computing: Motivating Parallelism, Scope of Parallel Computing, Parallel Programming Platforms: Implicit Parallelism: Trends in Microprocessor Architectures, Limitations of Memory System Performance, Dichotomy of Parallel Computing Platforms, Physical Organization of Parallel Platforms, Communication Costs in Parallel Machines, Routing Mechanisms for Interconnection Networks, Impact of Process-Processor Mapping and Mapping Techniques. T1: Ch: 1.1, 1.2, 2.1 – 2.7 RBT: L1, L2</p>	08
<p>Module – 2</p>	
<p>Principles of Parallel Algorithm Design: Preliminaries, Decomposition Techniques, Characteristics of Tasks and Interactions, Mapping Techniques for Load Balancing, Methods for Containing Interaction Overheads, Parallel Algorithm Models Basic Communication Operations: One-to-All Broadcast and All-to-One Reduction, Allto- All Broadcast and Reduction, All-Reduce and Prefix-Sum Operations, Scatter and Gather, All-to-All Personalized Communication, Circular Shift, Improving the Speed of Some Communication Operations T1: Ch 3, 4 RBT: L1, L2</p>	08
<p>Module – 3</p>	
<p>Analytical Modeling of Parallel Programs: Sources of Overhead in Parallel Programs, 08 Performance Metrics for Parallel Systems, The Effect of Granularity on Performance, Scalability of Parallel Systems. Minimum Execution Time and Minimum Cost-Optimal Execution Time, Asymptotic Analysis of Parallel Programs Section 5.7. Other Scalability Metrics, Programming Using the Message-Passing Paradigm: Principles of Message-Passing Programming, The Building Blocks: Send and Receive Operations, MPI: the Message Passing Interface, Topologies and Embedding, Overlapping Communication with Computation, Collective Communication and Computation Operations, Groups and Communicators T1: Ch 5, 6 RBT: L1, L2, L3</p>	
<p>Module – 4</p>	
<p>Programming Shared Address Space Platforms: Thread Basics, Why Threads?, The POSIX Thread API, Thread Basics: Creation and Termination, Synchronization Primitives in Pthreads, Controlling Thread and Synchronization Attributes, Thread Cancellation, Composite Synchronization Constructs, Tips for Designing Asynchronous Programs, OpenMP: a Standard for Directive Based Parallel Programming Dense Matrix Algorithms: Matrix-Vector Multiplication, Matrix-Matrix Multiplication, Solving a System of Linear Equations Sorting: Issues in Sorting on Parallel Computers, Sorting Networks, Bubble Sort and its Variants, Quicksort, Bucket and Sample Sort. T1: Ch 7, 8 9 RBT: L1, L2</p>	08
<p>Module – 5</p>	

Graph Algorithms: Definitions and Representation, Minimum Spanning Tree: Prim's 08 Algorithm, Single-Source Shortest Paths: Dijkstra's Algorithm, All-Pairs Shortest Paths, Transitive Closure, Connected Components, Algorithms for Sparse Graphs, Search Algorithms for Discrete Optimization Problems: Definitions and Examples, Sequential Search Algorithms, Search Overhead Factor, Parallel Depth-First Search, Parallel Best-First Search, Speedup, Anomalies in Parallel Search Algorithms T1: Ch10, 11 RBT: L1, L2	
Course outcomes: The students should be able to:	
<ul style="list-style-type: none"> • Illustrate the key factors affecting performance of CSE applications • Illustrate mapping of applications to high-performance computing systems • Apply hardware/software co-design for achieving performance on real-world applications 	
Question paper pattern:	
<ul style="list-style-type: none"> • 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 Parallel Computing, AnanthGrama, Anshul Gupta, George Karypis, and Vipin Kumar, 2nd edition, Addison-Welsey, 2003.	
Reference Books:	
<ol style="list-style-type: none"> 1. Grama, A. Gupta, G. Karypis, V. Kumar, An Introduction to Parallel Computing, Design and Analysis of Algorithms: 2/e, Addison-Wesley, 2003. 2. G.E. Karniadakis, R.M. Kirby II, Parallel Scientific Computing in C++ and MPI: A Seamless Approach to Parallel Algorithms and their Implementation, Cambridge University Press,2003. 3. Wilkinson and M. Allen, Parallel Programming: Techniques and Applications Using Networked Workstations and Parallel Computers, 2/E, Prentice Hall, 2005. 4. M.J. Quinn, Parallel Programming in C with MPI and OpenMP, McGraw-Hill, 2004. 5. G.S. Almasi and A. Gottlieb, Highly Parallel Computing, 2/E, Addison-Wesley, 1994. 6. David Culler Jaswinder Pal Singh,"Parallel Computer Architecture: A hardware/Software Approach", Morgan Kaufmann, 1999. 7. Kai Hwang, "Scalable Parallel Computing", McGraw Hill 1998. 	

ADVANCED COMPUTER ARCHITECTURES (Effective from the academic year 2018 -2019) SEMESTER - VIII			
Course Code	18CS733	CIE Marks	40
Number of Contact Hours/Week	3:0:0	SEE Marks	60
Total Number of Contact Hours	40	Exam Hours	03
CREDITS –3			
Course Learning Objectives: This course (18CS733) will enable students to:			

<ul style="list-style-type: none"> • Describe computer architecture. • Measure the performance of architectures in terms of right parameters. □ Summarize parallel architecture and the software used for them 	
Module 1	Contact Hours
<p>Theory of Parallelism: Parallel Computer Models, The State of Computing, Multiprocessors and Multicomputer, Multivector and SIMD Computers, PRAM and VLSI Models, Program and Network Properties, Conditions of Parallelism, Program Partitioning and Scheduling, Program Flow Mechanisms, System Interconnect Architectures, Principles of Scalable Performance, Performance Metrics and Measures, Parallel Processing Applications, Speedup Performance Laws. For all Algorithm or mechanism any one example is sufficient. Chapter 1 (1.1to 1.4), Chapter 2(2.1 to 2.4) Chapter 3 (3.1 to 3.3) RBT: L1, L2</p>	08
Module 2	
<p>Hardware Technologies 1: Processors and Memory Hierarchy, Advanced Processor Technology, Superscalar and Vector Processors, Memory Hierarchy Technology, Virtual Memory Technology. For all Algorithms or mechanisms any one example is sufficient. Chapter 4 (4.1 to 4.4) RBT: L1, L2, L3</p>	08
Module 3	
<p>Hardware Technologies 2: Bus Systems, Cache Memory Organizations, Shared Memory Organizations, Sequential and Weak Consistency Models, Pipelining and Superscalar Techniques, Linear Pipeline Processors, Nonlinear Pipeline Processors. For all Algorithms or mechanisms any one example is sufficient. Chapter 5 (5.1 to 5.4) Chapter 6 (6.1 to 6.2) RBT: L1, L2, L3</p>	08
Module 4	
<p>Parallel and Scalable Architectures: Multiprocessors and Multicomputers, Multiprocessor System Interconnects, Cache Coherence and Synchronization Mechanisms, MessagePassing Mechanisms, Multivector and SIMD Computers, Vector Processing Principles, Multivector Multiprocessors, Compound Vector Processing, Scalable, Multithreaded, and Dataflow Architectures, Latency-Hiding Techniques, Principles of Multithreading, Fine- Grain Multicomputers. For all Algorithms or mechanisms any one example is sufficient. Chapter 7 (7.1,7.2 and 7.4) Chapter 8(8.1 to 8.3) Chapter 9(9.1 to 9.3) RBT: L1, L2, L3</p>	08
Module 5	
<p>Software for parallel programming: Parallel Models, Languages, and Compilers ,Parallel Programming Models, Parallel Languages and Compilers, Dependence Analysis of Data Arrays. Instruction and System Level Parallelism, Instruction Level Parallelism, Computer Architecture, Contents, Basic Design Issues, Problem Definition, Model of a Typical</p>	08
<p>Processor, Compiler-detected Instruction Level Parallelism ,Operand Forwarding ,Reorder Buffer, Register Renaming ,Tomasulo’s Algorithm. For all Algorithms or mechanisms any one example is sufficient. Chapter 10(10.1 to 10.3) Chapter 12(12.1 to 12.9) RBT: L1, L2, L3</p>	
Course Outcomes: The student will be able to :	

<ul style="list-style-type: none"> • Explain the concepts of parallel computing and hardware technologies • Compare and contrast the parallel architectures • Illustrate parallel programming concepts
Question Paper Pattern:
<ul style="list-style-type: none"> • 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.
Textbooks:
1. Kai Hwang and Naresh Jotwani, Advanced Computer Architecture (SIE): Parallelism, Scalability, Programmability, McGraw Hill Education 3/e. 2015
Reference Books:
1. John L. Hennessy and David A. Patterson, Computer Architecture: A quantitative approach, 5th edition, Morgan Kaufmann Elseveir, 2013

USER INTERFACE DESIGN			
(Effective from the academic year 2018 -2019) SEMESTER			
- VII			
Course Code	18CS734	CIE Marks	40
Number of Contact Hours/Week	3:0:0	SEE Marks	60
Total Number of Contact Hours	40	Exam Hours	03
CREDITS –3			
Course Learning Objectives: This course (18CS734) will enable students to:			
<ul style="list-style-type: none"> • To study the concept of menus, windows, interfaces • To study about business functions • To study the characteristics and components of windows and the various controls for the windows. • To study about various problems in windows design with color, text, graphics and To study the testing methods 			
Module 1			Contact Hours
The User Interface-Introduction, Overview, The importance of user interface – Defining the user interface, The importance of Good design, Characteristics of graphical and web user interfaces, Principles of user interface design Textbook 1: Ch. 1,2 RBT: L1, L2			08
Module 2			
The User Interface Design process- Obstacles, Usability, Human characteristics in Design, Human Interaction speeds, Business functions-Business definition and requirement analysis, Basic business functions, Design standards. Textbook 1: Part-2 RBT: L1, L2			08

Module 3	
System menus and navigation schemes- Structures of menus, Functions of menus, Contents of menus, Formatting of menus, Phrasing the menu, Selecting menu choices, Navigating menus, Kinds of graphical menus. Textbook 1: Part-2 RBT: L1, L2	08
Module 4	
Windows - Characteristics, Components of window, Window presentation styles, Types of window, Window management, Organizing window functions, Window operations, Web systems, Characteristics of device based controls. Textbook 1: Part-2 RBT: L1, L2	08
Module 5	
Screen based controls- Operable control, Text control, Selection control, Custom control, Presentation control, Windows Tests-prototypes, kinds of tests. Textbook 1: Part-2 RBT: L1, L2	08
Course Outcomes: The student will be able to :	
<input type="checkbox"/> Design the User Interface, design, menu creation, windows creation and connection between menus and windows	
Question Paper Pattern:	
<ul style="list-style-type: none"> • The question paper will have ten questions. • Each full Question consisting of 20 marks 	
<ul style="list-style-type: none"> • 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. 	
Textbooks:	
1. Wilbert O. Galitz, “The Essential Guide to User Interface Design”, John Wiley & Sons, Second Edition 2002.	
Reference Books:	
<ol style="list-style-type: none"> 1. Ben Sheiderman, “Design the User Interface”, Pearson Education, 1998. 2. Alan Cooper, ”The Essential of User Interface Design”, Wiley- Dream Tech Ltd.,2002 	

DIGITAL IMAGE PROCESSING
(Effective from the academic year 2018 -2019) SEMESTER
- VII

Course Code	18CS741	CIE Marks	40
Number of Contact Hours/Week	3:0:0	SEE Marks	60
Total Number of Contact Hours	40	Exam Hours	03
CREDITS –3			
Course Learning Objectives: This course (18CS741) will enable students to:			
<ul style="list-style-type: none"> • Define the fundamental concepts in image processing • Evaluate techniques followed in image enhancements • Illustrate image segmentation and compression algorithms 			
Module 1			Contact Hours
Introduction Fundamental Steps in Digital Image Processing, Components of an Image Processing System, Sampling and Quantization, Representing Digital Images (Data structure), Some Basic Relationships Between Pixels- Neighbors and Connectivity of pixels in image, Examples of fields that uses digital mage processing Textbook 1: Ch.1.3 to 1.5, Ch. 2.4,2.5 RBT: L1, L2			08
Module 2			
Image Enhancement In The Spatial Domain: Some Basic Gray Level Transformations, Histogram Processing, Enhancement Using Arithmetic/Logic Operations, Basics of Spatial Filtering, Smoothing Spatial Filters, Sharpening Spatial Filters, Combining Spatial Enhancement Methods. Textbook 1: Ch.3 RBT: L1, L2, L3			08
Module 3			
Image Enhancement In Frequency Domain: Introduction, Fourier Transform, Discrete Fourier Transform (DFT), properties of DFT , Discrete Cosine Transform (DCT), Image filtering in frequency domain. Textbook 1: Ch.4.1,4.2 RBT: L1, L2, L3			08
Module 4			
Image Segmentation: Introduction, Detection of isolated points, line detection, Edge detection, Edge linking, Region based segmentation- Region growing, split and merge technique, local processing, regional processing, Hough transform, Segmentation using Threshold. Textbook 1: Ch.10.1 to 10.3 RBT: L1, L2, L3			08
Module 5			
Image Compression: Introduction, coding Redundancy , Inter-pixel redundancy, image compression model, Lossy and Lossless compression, Huffman Coding, Arithmetic Coding, LZW coding, Transform Coding, Sub-image size selection, blocking, DCT implementation using FFT, Run length coding. Textbook 1: Ch. 8.1 to 8.5 RBT: L1, L2, L3			08
Course Outcomes: The student will be able to :			

- Explain fundamentals of image processing
- Compare transformation algorithms

□ Contrast enhancement, segmentation and compression techniques

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.

Textbooks:

1. Rafael C G., Woods R E. and Eddins S L, Digital Image Processing, Prentice Hall, 2nd edition, 2008.

Reference Books:

1. Milan Sonka, "Image Processing, analysis and Machine Vision", Thomson Press India Ltd, Fourth Edition.
2. Fundamentals of Digital Image Processing- Anil K. Jain, 2nd Edition, Prentice Hall of India.
3. S. Sridhar , Digital Image Processing, Oxford University Press, 2nd Ed, 2016.
4. Digital Image Processing (with Matlab and Labview), Vipul singh, elsiver.Filip learning

NETWORK MANAGEMENT
(Effective from the academic year 2018 -2019) SEMESTER
– VII

Course Code	18CS742	CIE Marks	40
Number of Contact Hours/Week	3:0:0	SEE Marks	60
Total Number of Contact Hours	40	Exam Hours	03

CREDITS –3

Course Learning Objectives: This course (18CS742) will enable students to:

- Illustrate the need for interoperable network management.
- Explain the concepts and architecture behind standards based network management.
- Differentiate the concepts and terminology associated with SNMP and TMN
- Describe network management as a typical distributed application

Module 1	Contact Hours
<p>Introduction: Analogy of Telephone Network Management, Data and Telecommunication Network Distributed computing Environments, TCP/IP-Based Networks: The Internet and Intranets, Communications Protocols and Standards- Communication Architectures, Protocol Layers and Services; Case Histories of Networking and Management – The Importance of topology , Filtering Does Not Reduce Load on Node, Some Common Network Problems; Challenges of Information Technology Managers, Network Management: Goals, Organization, and Functions- Goal of Network Management, Network Provisioning, Network Operations and the NOC, Network Installation and Maintenance; Network and System Management, Network Management System platform, Current Status and Future of Network Management.</p> <p>Textbook 1: Ch.1 RBT: L1, L2</p>	08
<p>Module 2</p> <p>Basic Foundations: Standards, Models, and Language: Network Management Standards, Network Management Model, Organization Model, Information Model – Management Information Trees, Managed Object Perspectives, Communication Model; ASN.1- Terminology, Symbols, and Conventions, Objects and Data Types, Object Names, An Example of ASN.1 from ISO 8824; Encoding Structure; Macros, Functional Model. Textbook 1: Ch.3 RBT: L1, L2</p>	08
<p>Module 3</p> <p>SNMPv1 Network Management: Managed Network: The History of SNMP Management, Internet Organizations and standards, Internet Documents, The SNMP Model, The Organization Model, System Overview. The Information Model – Introduction, The Structure of Management Information, Managed Objects, Management Information Base. The SNMP Communication Model – The SNMP Architecture, Administrative Model, SNMP Specifications, SNMP Operations, SNMP MIB Group, Functional Model SNMP Management – RMON: Remote Monitoring, RMON SMI and MIB, RMON1- RMON1 Textual Conventions, RMON1 Groups and Functions, Relationship Between Control and Data Tables, RMON1 Common and Ethernet Groups, RMON Token Ring Extension Groups, RMON2 – The RMON2 Management Information Base, RMON2 Conformance Specifications.</p> <p>Textbook 1: Ch. 4,5, Ch.8 RBT: L1, L2</p>	08

Module 4	
<p>Broadband Access Networks, Broadband Access Technology; HFCT Technology: The Broadband LAN, The Cable Modem, The Cable Modem Termination System, The HFC Plant, The RF Spectrum for Cable Modem; Data Over Cable, Reference Architecture; HFC Management – Cable Modem and CMTS Management, HFC Link Management, RF Spectrum Management, DSL Technology; Asymmetric Digital Subscriber Line Technology – Role of the ADSL Access Network in an Overall Network, ADSL Architecture, ADSL Channeling Schemes, ADSL Encoding Schemes; ADSL Management – ADSL Network Management Elements, ADSL Configuration Management, ADSL Fault Management, ADSL Performance Management, SNMP-Based ADSL Line MIB, MIB Integration with Interfaces Groups in MIB-2, ADSL Configuration Profiles</p> <p>Textbook 1: Ch. 13 RBT: L1, L2</p>	08
Module 5	
<p>Network Management Applications: Configuration Management- Network Provisioning, Inventory Management, Network Topology, Fault Management- Fault Detection, Fault Location and Isolation 24 Techniques, Performance Management – Performance Metrics, Data Monitoring, Problem Isolation, Performance Statistics; Event Correlation Techniques – Rule-Based Reasoning, Model-Based Reasoning, CaseBased Reasoning, Codebook correlation Model, State Transition Graph Model, Finite State Machine Model, Security Management – Policies and Procedures, Security Breaches and the Resources Needed to Prevent Them, Firewalls, Cryptography, Authentication and Authorization, Client/Server Authentication Systems, Messages Transfer Security, Protection of Networks from Virus Attacks, Accounting Management, Report Management, Policy- Based Management, Service Level Management.</p> <p>Textbook 1: Ch.11 RBT: L1, L2</p>	08
Course Outcomes: The student will be able to :	
<ul style="list-style-type: none"> • Analyze the issues and challenges pertaining to management of emerging network technologies such as wired/wireless networks and high-speed internets. □ Apply network management standards to manage practical networks □ Formulate possible approaches for managing OSI network model. • Use on SNMP for managing the network • Use RMON for monitoring the behavior of the network • Identify the various components of network and formulate the scheme for the managing them 	
Question Paper Pattern:	
<ul style="list-style-type: none"> • 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. 	
Textbooks:	
1. Mani Subramanian: Network Management- Principles and Practice, 2nd Pearson Education, 2010.	
Reference Books:	
1. J. Richard Burke: Network management Concepts and Practices: a Hands-On Approach, PHI, 2008.	

NATURAL LANGUAGE PROCESSING
(Effective from the academic year 2018 -2019) SEMESTER
- VII

Course Code	18CS743	CIE Marks	40
Number of Contact Hours/Week	3:0:0	SEE Marks	60
Total Number of Contact Hours	40	Exam Hours	03
CREDITS –3			
Course Learning Objectives: This course (18CS743) will enable students to:			
Module – 1			Contact Hours
Overview and language modeling: Overview: Origins and challenges of NLP-Language and Grammar-Processing Indian Languages- NLP Applications-Information Retrieval. Language Modeling: Various Grammar- based Language Models-Statistical Language Model. Textbook 1: Ch. 1,2 RBT: L1, L2, L3			08
Module – 2			
Word level and syntactic analysis: Word Level Analysis: Regular Expressions-FiniteState Automata-Morphological Parsing-Spelling Error Detection and correction-Words and Word classes-Part-of Speech Tagging. Syntactic Analysis: Context-free GrammarConstituency- Parsing-Probabilistic Parsing. Textbook 1: Ch. 3,4 RBT: L1, L2, L3			08
Module – 3			
Extracting Relations from Text: From Word Sequences to Dependency Paths: Introduction, Subsequence Kernels for Relation Extraction, A Dependency-Path Kernel for Relation Extraction and Experimental Evaluation. Mining Diagnostic Text Reports by Learning to Annotate Knowledge Roles: Introduction, Domain Knowledge and Knowledge Roles, Frame Semantics and Semantic Role Labeling, Learning to Annotate Cases with Knowledge Roles and Evaluations. A Case Study in Natural Language Based Web Search: InFact System Overview, The GlobalSecurity.org Experience. Textbook 2: Ch. 3,4,5 RBT: L1, L2, L3			08
Module – 4			

<p>Evaluating Self-Explanations in iSTART: Word Matching, Latent Semantic Analysis, and Topic Models: Introduction, iSTART: Feedback Systems, iSTART: Evaluation of Feedback Systems,</p> <p>Textual Signatures: Identifying Text-Types Using Latent Semantic Analysis to Measure the Cohesion of Text Structures: Introduction, Cohesion, Coh-Metrix, Approaches to Analyzing Texts, Latent Semantic Analysis, Predictions, Results of Experiments.</p> <p>Automatic Document Separation: A Combination of Probabilistic Classification and Finite-State Sequence Modeling: Introduction, Related Work, Data Preparation, Document Separation as a Sequence Mapping Problem, Results.</p> <p>Evolving Explanatory Novel Patterns for Semantically-Based Text Mining: Related Work, A Semantically Guided Model for Effective Text Mining.</p> <p>Textbook 2: Ch. 6,7,8,9 RBT: L1, L2, L3</p>	08
Module – 5	
<p>INFORMATION RETRIEVAL AND LEXICAL RESOURCES: Information Retrieval: Design features of Information Retrieval Systems-Classical, Non classical, Alternative Models of Information Retrieval – valuation Lexical Resources: World Net-Frame Net-Stemmers-POS Tagger- Research Corpora.</p> <p>Textbook 1: Ch. 9,12 RBT: L1, L2, L3</p>	08
<p>Course outcomes: The students should be able to:</p>	
<ul style="list-style-type: none"> • Analyze the natural language text. • Define the importance of natural language. • Understand the concepts Text mining. • Illustrate information retrieval techniques. 	
<p>Question paper pattern:</p> <ul style="list-style-type: none"> • 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:	
<ol style="list-style-type: none"> 1. Tanveer Siddiqui, U.S. Tiwary, “Natural Language Processing and Information Retrieval”, Oxford University Press, 2008. 2. Anne Kao and Stephen R. Poteet (Eds), “Natural Language Processing and Text Mining”, Springer-Verlag London Limited 2007. 	
Reference Books:	
<ol style="list-style-type: none"> 1. Daniel Jurafsky and James H Martin, “Speech and Language Processing: An introduction to Natural Language Processing, Computational Linguistics and Speech Recognition”, 2nd Edition, Prentice Hall, 2008. 2. James Allen, “Natural Language Understanding”, 2nd edition, Benjamin/Cummings publishing company, 1995. 3. Gerald J. Kowalski and Mark.T. Maybury, “Information Storage and Retrieval systems”, Kluwer academic Publishers, 2000. 	

CRYPTOGRAPHY
(Effective from the academic year 2018 -2019) SEMESTER
- VII

Course Code	18CS744	CIE Marks	40
Number of Contact Hours/Week	3:0:0	SEE Marks	60
Total Number of Contact Hours	40	Exam Hours	03

CREDITS –3

Course Learning Objectives: This course (18CS744) will enable students to:

- Define cryptography and its principles
- Explain Cryptography algorithms
- Illustrate Public and Private key cryptography
- Explain Key management, distribution and certification
- Explain authentication protocols
- Tell about IPSec

Module – 1	Contact Hours
-------------------	----------------------

<p>Classical Encryption Techniques Symmetric Cipher Model, Cryptography, Cryptanalysis and Brute-Force Attack, Substitution Techniques, Caesar Cipher, Monoalphabetic Cipher, Playfair Cipher, Hill Cipher, Polyalphabetic Cipher, One Time Pad. Block Ciphers and the data encryption standard: Traditional block Cipher structure, stream Ciphers and block Ciphers, Motivation for the feistel Cipher structure, the feistel Cipher, The data encryption standard, DES encryption, DES decryption, A DES example, results, the avalanche effect, the strength of DES, the use of 56-Bit Keys, the nature of the DES algorithm, timing attacks, Block cipher design principles, number of rounds, design of function F, key schedule algorithm</p> <p>Textbook 1: Ch. 2.1,2.2, Ch. 3 RBT: L1, L2</p>	08
---	----

Module – 2	
-------------------	--

<p>Public-Key Cryptography and RSA: Principles of public-key cryptosystems. Public-key cryptosystems. Applications for public-key cryptosystems, requirements for public-key cryptosystems. public-key cryptanalysis. The RSA algorithm, description of the algorithm, computational aspects, the security of RSA.</p> <p>Other Public-Key Cryptosystems: Diffie-hellman key exchange, The algorithm, key exchange protocols, man in the middle attack, Elgamal Cryptographic systems Textbook 1: Ch. 9, Ch. 10.1,10.2 RBT: L1, L2</p>	08
---	----

Module – 3	
-------------------	--

<p>Elliptic curve arithmetic, abelian groups, elliptic curves over real numbers, elliptic curves over Z_p, elliptic curves over $GF(2^m)$, Elliptic curve cryptography, Analog of Diffie-hellman key exchange, Elliptic curve encryption/ decryption, security of Elliptic curve cryptography, Pseudorandom number generation based on an asymmetric cipher, PRNG based on RSA.</p> <p>Key Management and Distribution: Symmetric key distribution using Symmetric encryption, A key distribution scenario, Hierarchical key control, session key lifetime, a transparent key control scheme, Decentralized key control, controlling key usage, Symmetric key distribution using asymmetric encryption, simple secret key distribution, secret key distribution with confidentiality and authentication, A hybrid scheme, distribution of public keys, public announcement of public keys, publicly available directory, public key</p>	08
--	----

authority, public keys certificates. Textbook 1: Ch. 10.3-10.5, Ch.14.1 to 14.3 RBT: L1, L2	
Module – 4	
X-509 certificates. Certificates, X-509 version 3, public key infrastructure . User Authentication: Remote user Authentication principles, Mutual Authentication, one way Authentication, remote user Authentication using Symmetric encryption, Mutual Authentication, one way Authentication, Kerberos, Motivation , Kerberos version 4, Kerberos version 5, Remote user Authentication using Asymmetric encryption, Mutual Authentication, one way Authentication. Electronic Mail Security: Pretty good privacy, notation, operational; description, S/MIME, RFC5322, Multipurpose internet mail extensions, S/MIME functionality, S/MIME messages, S/MIME certificate processing, enhanced security services, Domain keys identified mail, internet mail architecture, E-Mail threats, DKIM strategy, DKIM functional flow. Textbook 1: Ch. 14.4, Ch. 15.1 to 15.4, Ch.19 RBT: L1, L2	08
Module – 5	
IP Security: IP Security overview, applications of IPsec, benefits of IPsec, Routing applications, IPsec documents, IPsec services, transport and tunnel modes, IP Security policy, Security associations, Security associations database, Security policy database, IP traffic processing, Encapsulating Security payload, ESP format, encryption and authentication algorithms, Padding, Anti replay service Transport and tunnel modes, combining security associations, authentication plus confidentiality, basic combinations of security associations, internet key exchange, key determinations protocol, header and payload formats, cryptographic suits. Textbook 1: Ch. 20.1 to 20.3 RBT: L1, L2	08
Course outcomes: The students should be able to:	
<ul style="list-style-type: none"> • Define cryptography and its principles • Explain Cryptography algorithms • Illustrate Public and Private key cryptography • Explain Key management, distribution and certification • Explain authentication protocols • Tell about IPSec 	
Question paper pattern:	
<ul style="list-style-type: none"> • 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, Pearson 6 th edition.	
Reference Books:	
1. V K Pachhare: Cryptography and Information Security, PHI 2 nd Edition.	



ವಿಶ್ವೇಶ್ವರಯ್ಯ ತಾಂತ್ರಿಕ ವಿಶ್ವವಿದ್ಯಾಲಯ

"ವಿಜ್ಞಾನಂ ಅಧಿನಿಯಮಂ ರಕ್ಷಿಸತು" ಅರ್ಥದಲ್ಲಿ, ಕರ್ನಾಟಕ ಸರ್ಕಾರದಿಂದ ಸ್ಥಾಪಿತವಾದ ರಾಜ್ಯ ವಿಶ್ವವಿದ್ಯಾಲಯ
"ಜ್ಞಾನ ಸಂಗಮ", ಬೆಳಗಾವಿ-೫೯೦೦೧೮, ಕರ್ನಾಟಕ, ಭಾರತ

Visvesvaraya Technological University

(State University of Government of Karnataka Established as per the VTU Act, 1994)

"Jnana Sangama" Belagavi-590018, Karnataka, India
Phone: (0831) 2498100, Fax: (0831) 2405467, Website: vtu.ac.in

Dr. A. S. Deshpande B.E., M.Tech., Ph.D.

Registrar

Phone: (0831) 2498100

Fax: (0831) 2405467

Ref: VTU/BGM/BOS/A9/2020-21 / 2749

Date: 23 SEP 2021

CIRCULAR

Subject: Updated syllabus of 18CS745 regarding...

Reference:

1. Approval of Chairperson BoS in CSE dated 08.09.2021
2. Approval of Hon'ble Vice-Chancellor, dated: 13.09.2021

This is to inform all concerned that the Professional Elective Course "**Robotic Process Automation Design & Development (18CS745)**" in Computer Science and Engineering program has been modified to map with chapter contents of the prescribed textbook. The updated syllabus copy has been enclosed with the circular for kind reference to the stakeholders.

The principals of all the Engineering Colleges coming under the ambit of the University are hereby informed to bring the updated syllabus of 18CS745 to the notice of the faculty and students of the CSE / department of your college.

Encl: As mentioned above.

To,

The Principals of all affiliated/ constituent /Autonomous Engineering Colleges, under the ambit of VTU Belagavi.

Copy to.

1. To the Hon'ble Vice-Chancellor through the secretary to VC, VTU Belagavi for information
2. The Registrar (Evaluation), VTU Belagavi for information.
3. The Regional Directors (I/c) of all the regional offices of VTU for circulation.
4. The Director ITI SMU CNC Belagavi for uploading on VTU website
5. PS to Registrar VTU Belagavi
6. All the concerned Special Officer/s and Caseworker/s of the academic section, VTU, Belagavi

23-9-2021
REGISTRAR
VSK

ROBOTIC PROCESS AUTOMATION DESIGN & DEVELOPMENT

(Effective from the academic year 2018-2019)

SEMESTER-VII

CourseCode	18CS745	CIEMarks	40
NumberofContactHours/Week	3:0:0	SEEMarks	60
TotalNumberofContactHours	40	ExamHours	3Hrs
		CREDITS	03

Course Learning Objectives: This course(18CS745) will enable students to:

1. To understand basic concepts of RPA
2. To Describe RPA, where it can be applied and how it implemented
3. To Describe the different types of variables, Control Flow and data manipulation techniques
4. To Understand Image,Text and Data Tables Automation
5. To Describe various types of Exceptions and strategies to handle

Module-1

**Contact
Hours**

RPA Foundations- What is RPA – Flavors of RPA- History of RPA- The 08 Benefits of RPA- The downsides of RPA- RPA Compared to BPO, BPM and BPA – Consumer Willingness for Automation- The Workforce of the Future- RPA Skills-On-Premise Vs. the Cloud- Web Technology- Programming Languages and Low Code- OCR-Databases-APIs- AI-Cognitive Automation-Agile, Scrum, Kanban and Waterfall0 DevOps- Flowcharts.

Textbook 1: Ch 1, Ch 2, RBT:L1,L2

Module-2

RPA Platforms- Components of RPA- RPA Platforms-About Ui Path- About 08 UiPath - The future of automation - Record and Play - Downloading and installing UiPath Studio -Learning Ui Path Studio- - Task recorder - Step-bystep examples using the recorder. **Textbook 2: Ch 1, Ch 2, RBT: L1, L2**

Module-3

Sequence, Flowchart, and Control Flow-Sequencing the workflow- 08 Activities-Control flow, various types of loops, and decision making-Step-bystep example using Sequence and Flowchart-Step-by-step example using Sequence and Control flow-Data Manipulation-Variables and ScopeCollections-Arguments – Purpose and use-Data table usage with examplesClipboard management-File operation with step-by-step example- CSV/Excel to data table and vice versa (with a step-by-stepexample).

Textbook 2: Ch 3, Ch 4, RBT:L1,L2

Module-4

Taking Control of the Controls- Finding and attaching windows- Finding the 08 control- Techniques for waiting for a control- Act on controls – mouse and keyboard activities- Working with UiExplorer- Handling events- Revisit recorder- Screen Scraping- When to use OCR- Types of OCR available- How to use OCR- Avoiding typical failure points.

Text book 2: Ch 5 RBT:L1,L2

Module-5

Exception Handling, Debugging, and Logging- Exception handling- Common 08 exceptions and ways to handle them- Logging and taking screenshots- Debugging techniques- Collecting crash dumps- Error reporting- Future of RPA

Text book 2: Ch 8 Text book 1: Ch 13 RBT:L1,L2

Courseoutcomes:Thestudentsshouldbeableto:

- To Understand the basic concepts of RPA
- To Describevarious components and platforms of RPA
- To Describe the different types of variables, control flow and data manipulation techniques
- To Understand various control techniques and OCR in RPA
- ToDescribevarious types andstrategies tohandle exceptions

Questionpaperpattern:

- The question paper will have tenquestions.
- There will be 2questions 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. Tom Taulli, The Robotic Process Automation Handbook : A Guide to Implementing RPA Systems,2020, ISBN-13 (electronic): 978-1-4842-5729-6, Publisher : A press
2. Alok Mani Tripathi, Learning Robotic Process Automation, Publisher: Packt Publishing Release Date: March 2018 ISBN: 9781788470940

Reference Books:

1. Frank Casale, Rebecca Dilla, Heidi Jaynes ,Lauren Livingston,“Introduction to Robotic Process Automation: a Primer”, Institute of Robotic Process Automation.
2. Richard Murdoch, Robotic Process Automation: Guide To Building Software Robots, Automate Repetitive Tasks & Become An RPA Consultant
3. Srikanth Merianda, Robotic Process Automation Tools, Process Automation and their benefits: Understanding RPA and Intelligent Automation
4. <https://www.uipath.com/rpa/robotic-process-automation>

INTRODUCTION TO BIG DATA ANALYTICS
(OPEN ELECTIVE)
(Effective from the academic year 2018 -2019) SEMESTER
- VII

Course Code	18CS751	CIE Marks	40
Number of Contact Hours/Week	3:0:0	SEE Marks	60
Total Number of Contact Hours	40	Exam Hours	03
CREDITS –3			
Course Learning Objectives: This course (18CS751) will enable students to:			
<ul style="list-style-type: none"> • Interpret the data in the context of the business. • Identify an appropriate method to analyze the data • Show analytical model of a system 			
Module – 1			Teaching Hours
<p>Introduction to Data Analytics and Decision Making: Introduction, Overview of the Book, The Methods, The Software, Modeling and Models, Graphical Models, Algebraic Models, Spreadsheet Models, Seven-Step Modeling Process.Describing the Distribution of a Single Variable:Introduction,Basic Concepts, Populations and Samples, Data Sets,Variables,and Observations, Types of Data, Descriptive Measures for Categorical Variables, Descriptive Measures for Numerical Variables, Numerical Summary Measures, Numerical Summary Measures with StatTools,Charts for Numerical Variables, Time Series Data, Outliers and Missing Values,Outliers,Missing Values, Excel Tables for Filtering,Sorting,and Summarizing.</p> <p>Finding Relationships among Variables: Introduction, Relationships among Categorical Variables, Relationships among Categorical Variables and a Numerical Variable, Stacked and Unstacked Formats, Relationships among Numerical Variables, Scatterplots, Correlation and Covariance, Pivot Tables.</p> <p>Textbook 1: Ch. 1,2,3 RBT: L1, L2, L3</p>			08
Module – 2			
<p>Probability and Probability Distributions:Introduction,Probability Essentials, Rule of Complements, Addition Rule, Conditional Probability and the Multiplication Rule, Probabilistic Independence, Equally Likely Events, Courseive Versus Objective Probabilities, Probability Distribution of a Single Random Variable, Summary Measures of a Probability Distribution, Conditional Mean and Variance, Introduction to Simulation.</p> <p>Normal,Binormal,Poisson,and Exponential Distributions:Introduction,The Normal Distribution, Continuous Distributions and Density Functions, The Normal Density,Standardizing:Z-Values,Normal Tables and Z-Values, Normal Calculations in Excel, Empirical Rules Revisited, Weighted Sums of Normal Random Variables, Applications of the Normal Random Distribution, The Binomial Distribution, Mean and Standard Deviation of the Binomial Distribution, The Binomial Distribution in the Context of Sampling, The Normal Approximation to the Binomial, Applications of the Binomial Distribution, The Poisson and Exponential Distributions, The Poisson Distribution, The Exponential Distribution.</p> <p>Textbook 1: Ch. 4,5 RBT: L1, L2, L3</p>			08
Module – 3			
Decision Making under Uncertainty: Introduction,Elements of Decision Analysis, Payoff			08

<p>Tables, Possible Decision Criteria, Expected Monetary Value(EMY),Sensitivity Analysis, Decision Trees, Risk Profiles, The Precision Tree Add-In,Bayes' Rule, Multistage Decision Problems and the Value of Information, The Value of Information, Risk Aversion and Expected Utility, Utility Functions, Exponential Utility, Certainty Equivalents, Is Expected Utility Maximization Used?</p> <p>Sampling and Sampling Distributions: Introduction, Sampling Terminology, Methods for Selecting Random Samples, Simple Random Sampling, Systematic Sampling, Stratified Sampling, Cluster Sampling, Multistage Sampling Schemes, Introduction to Estimation, Sources of Estimation Error, Key Terms in Sampling, Sampling Distribution of the Sample Mean, The Central Limit Theorem, Sample Size Selection, Summary of Key Ideas for Simple Random Sampling.</p> <p>Textbook 1: Ch. 6,7 RBT: L1, L2, L3</p>	
<p>Module – 4</p>	
<p>Confidence Interval Estimation: Introduction, Sampling Distributions, The t Distribution, Other Sampling Distributions, Confidence Interval for a Mean, Confidence Interval for a Total, Confidence Interval for a Proportion, Confidence Interval for a Standard Deviation, Confidence Interval for the Difference between Means, Independent Samples, Paired Samples, Confidence Interval for the Difference between Proportions, Sample Size Selection, Sample Size Selection for Estimation of the Mean, Sample Size Selection for Estimation of Other Parameters.</p> <p>Hypothesis Testing:Introduction,Concepts in Hypothesis Testing, Null and Alternative Hypothesis, One-Tailed Versus Two-Tailed Tests, Types of Errors, Significance Level and Rejection Region, Significance from p-values, Type II Errors and Power, Hypothesis Tests and Confidence Intervals, Practical versus Statistical Significance, Hypothesis Tests for a Population Mean, Hypothesis Tests for Other Parameters, Hypothesis Tests for a Population Proportion, Hypothesis Tests for Differences between Population Means, Hypothesis Test for Equal Population Variances, Hypothesis Tests for Difference between Population Proportions, Tests for Normality, Chi-Square Test for Independence. Textbook 1: Ch. 8,9 RBT: L1, L2, L3</p>	<p>08</p>
<p>Module – 5</p>	
<p>Regression Analysis: Estimating Relationships: Introduction, Scatterplots : Graphing Relationships, Linear versus Nonlinear Relationships,Outliers,Unequal Variance, No Relationship,Correlations:Indications of Linear Relationships, Simple Linear Regression, Least Squares Estimation, Standard Error of Estimate, The Percentage of Variation Explained:R-Square,Multiple Regression, Interpretation of Regression Coefficients, Interpretation of Standard Error of Estimate and R-Square, Modeling Possibilities, Dummy Variables, Interaction Variables, Nonlinear Transformations, Validation of the Fit.</p> <p>Regression Analysis: Statistical Inference:Introduction,The Statistical Model, Inferences About the Regression Coefficients, Sampling Distribution of the Regression Coefficients, Hypothesis Tests for the Regression Coefficients and p-Values, A Test for the Overall Fit: The ANOVA Table,Multicollinearity,Include/Exclude Decisions, Stepwise Regression,Outliers,Violations of Regression Assumptions,Nonconstant Error Variance,Nonnormality of Residuals,Autocorrelated Residuals ,Prediction.</p> <p>Textbook 1: Ch. 10,11 RBT: L1, L2, L3</p>	
<p>Course outcomes: The students should be able to:</p>	
<ul style="list-style-type: none"> • Explain the importance of data and data analysis • Interpret the probabilistic models for data 	
<ul style="list-style-type: none"> • Define hypothesis, uncertainty principle • Evaluate regression analysis 	
<p>Question Paper Pattern:</p>	

- 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 Books:

1. S C Albright and W L Winston, Business analytics: data analysis and decision making, 5/e Cenage Learning

Reference Books:

1. ArshdeepBahga, Vijay Madiseti, "Big Data Analytics: A Hands-On Approach", 1st Edition, VPT Publications, 2018. ISBN-13: 978-0996025577
2. Raj Kamal and Preeti Saxena, "Big Data Analytics Introduction to Hadoop, Spark, and MachineLearning", McGraw Hill Education, 2018 ISBN: 9789353164966, 9353164966

PYTHON APPLICATION PROGRAMMING
(OPEN ELECTIVE)
(Effective from the academic year 2018 -2019) SEMESTER
- VI

Course Code	18CS752	IA Marks	40
Number of Lecture Hours/Week	3:0:0	Exam Marks	60
Total Number of Lecture Hours	40	Exam Hours	03

CREDITS – 03

Course Learning Objectives: This course (18CS752) will enable students to
<ul style="list-style-type: none"> • Learn Syntax and Semantics and create Functions in Python. • Handle Strings and Files in Python. • Understand Lists, Dictionaries and Regular expressions in Python. • Implement Object Oriented Programming concepts in Python • Build Web Services and introduction to Network and Database Programming in Python.

Module – 1	Teaching Hours
Why should you learn to write programs, Variables, expressions and statements, Conditional execution, Functions Textbook 1: Chapters 1 – 4 RBT: L1, L2, L3	08

Module – 2	Teaching Hours
Iteration, Strings, Files Textbook 1: Chapters 5– 7 RBT: L1, L2, L3	08

Module – 3	Teaching Hours
Lists, Dictionaries, Tuples, Regular Expressions Textbook 1: Chapters 8 - 11 RBT: L1, L2, L3	08

Module – 4	Teaching Hours
Classes and objects, Classes and functions, Classes and methods Textbook 2: Chapters 15 – 17 RBT: L1, L2, L3	08

Module – 5	Teaching Hours
Networked programs, Using Web Services, Using databases and SQL Textbook 1: Chapters 12– 13, 15 RBT: L1, L2, L3	08

Course Outcomes: After studying this course, students will be able to
<ul style="list-style-type: none"> • 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.

Question paper pattern:
<ul style="list-style-type: none"> • 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 Books:

1. Charles R. Severance, "**Python for Everybody: Exploring Data Using Python 3**", 1st Edition, CreateSpace Independent Publishing Platform, 2016. (http://do1.drchuck.com/pythonlearn/EN_us/pythonlearn.pdf)
2. Allen B. Downey, "**Think Python: How to Think Like a Computer Scientist**", 2nd Edition, Green Tea Press, 2015. (<http://greenteapress.com/thinkpython2/thinkpython2.pdf>) (Download pdf files from the above links)

Reference Books:

1. Charles Dierbach, "**Introduction to Computer Science Using Python**", 1st Edition, Wiley India Pvt Ltd, 2015. ISBN-13: 978-8126556014
2. Gowrishankar S, Veena A, "**Introduction to Python Programming**", 1st Edition, CRC Press/Taylor & Francis, 2018. ISBN-13: 978-0815394372
3. Mark Lutz, "**Programming Python**", 4th Edition, O'Reilly Media, 2011. ISBN-13: 9789350232873
4. Roberto Tamassia, Michael H Goldwasser, Michael T Goodrich, "**Data Structures and Algorithms in Python**", 1st Edition, Wiley India Pvt Ltd, 2016. ISBN-13: 978-8126562176
5. Reema Thareja, "**Python Programming Using Problem Solving Approach**", Oxford university press, 2017. ISBN-13: 978-0199480173

INTRODUCTION TO ARTIFICIAL INTELLIGENCE
(OPEN ELECTIVE)
(Effective from the academic year 2018 -2019) SEMESTER
- VII

Course Code	18CS753	CIE Marks	40
Number of Contact Hours/Week	3:0:0	SEE Marks	60
Total Number of Contact Hours	40	Exam Hours	03
CREDITS –3			
Course Learning Objectives: This course (18CS753) will enable students to:			
<input type="checkbox"/> Identify the problems where AI is required and the different methods available <input type="checkbox"/> Compare and contrast different AI techniques available. <input type="checkbox"/> Define and explain learning algorithms			
Module – 1			Teaching Hours
What is artificial intelligence?, Problems, Problem Spaces and search TextBook1: Ch 1, 2 RBT: L1, L2			08
Module – 2			
Knowledge Representation Issues, Using Predicate Logic, Representing knowledge using Rules, TextBook1: Ch 4, 5 and 6. RBT: L1, L2			08
Module – 3			
Symbolic Reasoning under Uncertainty, Statistical reasoning TextBook1: Ch 7, 8 RBT: L1, L2			08
Module – 4			
Game Playing, Natural Language Processing TextBook1: Ch 12 and 15 RBT: L1, L2			08
Module – 5			
Learning, Expert Systems. TextBook1: Ch 17 and 20 RBT: L1, L2			08
Course outcomes: The students should be able to:			
<ul style="list-style-type: none"> • Identify the AI based problems • Apply techniques to solve the AI problems • Define learning and explain various learning techniques <input type="checkbox"/> Discuss on expert systems 			
Question paper pattern:			
<ul style="list-style-type: none"> • 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 Books:			
1. E. Rich , K. Knight & S. B. Nair - Artificial Intelligence, 3/e, McGraw Hill.			
Reference Books:			

1. Artificial Intelligence: A Modern Approach, Stuart Rusell, Peter Norving, Pearson Education 2nd Edition.
2. Dan W. Patterson, Introduction to Artificial Intelligence and Expert Systems – Prentice Hal of India.
3. G. Luger, “Artificial Intelligence: Structures and Strategies for complex problem Solving”, Fourth Edition, Pearson Education, 2002.
4. Artificial Intelligence and Expert Systems Development by D W Rolston-Mc Graw hill.
5. N.P. Padhy “Artificial Intelligence and Intelligent Systems” , Oxford University Press-2015

INTRODUCTION TO DOT NET FRAMEWORK FOR APPLICATION DEVELOPMENT
(OPEN ELECTIVE)
(Effective from the academic year 2018 -2019) SEMESTER
- VII

Course Code	18CS754	CIE Marks	40
Number of Contact Hours/Week	3:0:0	SEE Marks	60
Total Number of Contact Hours	40	Exam Hours	03
CREDITS –3			
Course Learning Objectives: This course (18CS754) will enable students to:			
<ul style="list-style-type: none"> • Inspect Visual Studio programming environment and toolset designed to build applications for Microsoft Windows • Understand Object Oriented Programming concepts in C# programming language. • Interpret Interfaces and define custom interfaces for application. • Build custom collections and generics in C# • Construct events and query data using query expressions 			
Module – 1			Teaching Hours
Introducing Microsoft Visual C# and Microsoft Visual Studio 2015: Welcome to C#, Working with variables, operators and expressions, Writing methods and applying scope, Using decision statements, Using compound assignment and iteration statements, Managing errors and exceptions T1: Chapter 1 – Chapter 6 RBT: L1, L2			08
Module – 2			
Understanding the C# object model: Creating and Managing classes and objects, Understanding values and references, Creating value types with enumerations and structures, Using arrays Textbook 1: Ch 7 to 10 RBT: L1, L2			08
Module – 3			
Understanding parameter arrays, Working with inheritance, Creating interfaces and defining abstract classes, Using garbage collection and resource management Textbook 1: Ch 11 to 14 RBT: L1, L2			08
Module – 4			
Defining Extensible Types with C#: Implementing properties to access fields, Using indexers, Introducing generics, Using collections Textbook 1: Ch 15 to 18 RBT: L1, L2			08
Module – 5			
Enumerating Collections, Decoupling application logic and handling events, Querying inmemory data by using query expressions, Operator overloading Textbook 1: Ch 19 to 22 RBT: L1, L2			08
Course outcomes: The students should be able to:			
<ul style="list-style-type: none"> • Build applications on Visual Studio .NET platform by understanding the syntax and semantics of C# • Demonstrate Object Oriented Programming concepts in C# programming language 			

- Design custom interfaces for applications and leverage the available built-in interfaces in building complex applications.
- Illustrate the use of generics and collections in C#
- Compose queries to query in-memory data and define own operator behaviour

Question paper pattern:

The question paper will have TEN questions.

There will be TWO questions from each module.

Each question will have questions covering all the topics under a module.

The students will have to answer FIVE full questions, selecting ONE full question from each module.

Text Books:

1. John Sharp, Microsoft Visual C# Step by Step, 8th Edition, PHI Learning Pvt. Ltd. 2016

Reference Books:

1. Christian Nagel, "C# 6 and .NET Core 1.0", 1st Edition, Wiley India Pvt Ltd, 2016. Andrew Stellman and Jennifer Greene, "Head First C#", 3rd Edition, O'Reilly Publications, 2013.
2. Mark Michaelis, "Essential C# 6.0", 5th Edition, Pearson Education India, 2016.
3. Andrew Troelsen, "Prof C# 5.0 and the .NET 4.5 Framework", 6th Edition, Apress and Dreamtech Press, 2012.

ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING LABORATORY
(Effective from the academic year 2018 -2019) SEMESTER
- VII

Course Code	18CSL76	CIE Marks	40
Number of Contact Hours/Week	0:0:2	SEE Marks	60
Total Number of Lab Contact Hours	36	Exam Hours	03

Credits – 2

Course Learning Objectives: This course (18CSL76) will enable students to:

- Implement and evaluate AI and ML algorithms in and Python programming language.

Descriptions (if any):

Installation procedure of the required software must be demonstrated, carried out in groups and documented in the journal.

Programs List:

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 to classify 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 Regression algorithm in order to fit data points. Select appropriate data set for your experiment and draw graphs

Laboratory Outcomes: The student should be able to:

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

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 accordance with university regulations*)
 - q) For laboratories having only one part – Procedure + Execution + Viva-Voce: 15+70+15 = 100 Marks
 - r) For laboratories having PART A and PART B
 - i. Part A – Procedure + Execution + Viva = 6 + 28 + 6 = 40 Marks
 - ii. Part B – Procedure + Execution + Viva = 9 + 42 + 9 = 60 Marks

INTERNET OF THINGS
(Effective from the academic year 2018 -2019) SEMESTER
- VIII

Course Code	18CS81	CIE Marks	40
Number of Contact Hours/Week	3:0:0	SEE Marks	60
Total Number of Contact Hours	40	Exam Hours	03
CREDITS –3			
Course Learning Objectives: This course (18CS81) will enable students to:			
<ul style="list-style-type: none"> • Assess the genesis and impact of IoT applications, architectures in real world. • Illustrate diverse methods of deploying smart objects and connect them to network. • Compare different Application protocols for IoT. • Infer the role of Data Analytics and Security in IoT. • Identifysensor technologies for sensing real world entities and understand the role of IoT in various domains of Industry. 			
Module 1			Contact Hours
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. Textbook 1: Ch.1, 2 RBT: L1, L2, L3			08
Module 2			
Smart Objects: The “Things” in IoT, Sensors, Actuators, and Smart Objects, Sensor Networks, Connecting Smart Objects, Communications Criteria, IoT Access Technologies. Textbook 1: Ch.3, 4 RBT: L1, L2, L3			08
Module 3			
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. Textbook 1: Ch.5, 6 RBT: L1, L2, L3			08
Module 4			
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 Textbook 1: Ch.7, 8 RBT: L1, L2, L3			08
Module 5			
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			08

Strategy for Smarter Cities, Smart City IoT Architecture, Smart City Security Architecture, Smart City Use-Case Examples. Textbook 1: Ch.12 Textbook 2: Ch.7.1 to 7.4, Ch.8.1 to 8.4, 8.6 RBT: L1, L2, L3	
Course Outcomes: The student will be able to :	
<ul style="list-style-type: none"> • 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:	
<ul style="list-style-type: none"> • 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. 	
Textbooks:	
<ol style="list-style-type: none"> 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: 978-9386873743) 2. Srinivasa K G, "Internet of Things", CENGAGE Learning India, 2017 	
Reference Books:	
<ol style="list-style-type: none"> 1. Vijay Madiseti 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) 	
Mandatory Note:	
Distribution of CIE Marks is as follows (Total 40 Marks):	
<ul style="list-style-type: none"> • 20 Marks through IA Tests • 20 Marks through practical assessment 	
Maintain a copy of the report for verification during LIC visit.	
Possible list of practicals:	
<ol style="list-style-type: none"> 1. Transmit a string using UART 2. Point-to-Point communication of two Motes over the radio frequency. 3. Multi-point to single point communication of Motes over the radio frequency.LAN (Subnetting). 4. I2C protocol study 5. Reading Temperature and Relative Humidity value from the sensor 	

MOBILE COMPUTING			
(Effective from the academic year 2018 -2019) SEMESTER			
- VIII			
Course Code	18CS821	CIE Marks	40
Number of Contact Hours/Week	3:0:0	SEE Marks	60
Total Number of Contact Hours	40	Exam Hours	03
CREDITS -3			

<p>Course Learning Objectives: This course (18CS821) will enable students to:</p> <ul style="list-style-type: none"> • 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 	
Module 1	Contact Hours
<p>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</p> <p>Textbook1: 2.4 - 2.6, 4.4 - 4.6, 5, 6. RBT: L1, L2</p>	08
Module 2	
<p>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.</p> <p>Textbook 1: 7,9.2 - 9.7, 12.2 - 12.6 RBT: L1, L2</p>	08
Module 3	
<p>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 Textbook 2: 7, 8.</p> <p>RBT: L1, L2</p>	08
Module 4	
<p>Building Wireless Internet Applications: Thin client overview: Architecture, the client, Middleware, messaging Servers, Processing a Wireless request, Wireless Applications</p> <p>Protocol (WAP) Overview, Wireless Languages: Markup Languages, HDML, WML, 10 Hours HTML, cHTML, XHTML, VoiceXML.</p> <p>Textbook 2: 11, 12, 13 RBT: L1, L2</p>	08
Module 5	
<p>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.</p> <p>Textbook 1: 15.1 - 15.10 RBT: L1, L2</p>	08

Course Outcomes: The student will be able to :
The students shall able to: <ul style="list-style-type: none"> • Explain state of art techniques in wireless communication. • Discover CDMA, GSM, Mobile IP, Wimax • Demonstrate program for CLDC, MIDP let model and security concerns
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: <ol style="list-style-type: none"> 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: <ol style="list-style-type: none"> 1. Raj kamal: Mobile Computing, Oxford University Press, 2007. 2. Iti Saha Misra: Wireless Communications and Networks, 3G and Beyond, Tata McGraw Hill, 2009.

STORAGE AREA NETWORKS (Effective from the academic year 2018 -2019) SEMESTER - VII			
Course Code	18CS822	CIE Marks	40
Number of Contact Hours/Week	3:0:0	SEE Marks	60
Total Number of Contact Hours	40	Exam Hours	03
CREDITS –3			
Course Learning Objectives: This course (18CS822) will enable students to: <ul style="list-style-type: none"> • 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			Contact 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
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

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
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
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 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
Course Outcomes: The student will be able to :		
<ul style="list-style-type: none"> 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 		
Question Paper Pattern:		
<ul style="list-style-type: none"> 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. 		
Textbooks:		
1. EMC Education Services, “Information Storage and Management” , Wiley India Publications, 2009. ISBN: 9781118094839		
Reference Books:		
1. Paul Massiglia, Richard Barker, "Storage Area Network Essentials: A Complete Guide to Understanding and Implementing SANs Paperback" , 1st Edition, Wiley India Publications, 2008		

NOSQL DATABASE			
(Effective from the academic year 2018 -2019) SEMESTER			
- VIII			
Course Code	18CS823	CIE Marks	40
Number of Contact Hours/Week	3:0:0	SEE Marks	60
Total Number of Contact Hours	40	Exam Hours	03

CREDITS –3

Course Learning Objectives: This course (18CS823) will enable students to:

- Define, compare and use the four types of NoSQL Databases (Document-oriented, Key-Value Pairs, Column-oriented and Graph).
- Demonstrate an understanding of the detailed architecture, define objects, load data, query data and performance tune Column-oriented NoSQL databases.
- Explain the detailed architecture, define objects, load data, query data and performance tune Document-oriented NoSQL databases.

Module 1	Contact Hours
<p>Why NoSQL? The Value of Relational Databases, Getting at Persistent Data, Concurrency, Integration, A (Mostly) Standard Model, Impedance Mismatch, Application and Integration Databases, Attack of the Clusters, The Emergence of NoSQL, Aggregate Data Models; Aggregates, Example of Relations and Aggregates, Consequences of Aggregate Orientation, Key-Value and Document Data Models, Column-Family Stores, Summarizing Aggregate-Oriented Databases. More Details on Data Models; Relationships, Graph Databases, Schemaless Databases, Materialized Views, Modeling for Data Access, Textbook1: Chapter 1,2,3 RBT: L1, L2, L3</p>	08
<p>Module 2</p> <p>Distribution Models; Single Server, Sharding, Master-Slave Replication, Peer-to-Peer Replication, Combining Sharding and Replication. Consistency, Update Consistency, Read Consistency, Relaxing Consistency, The CAP Theorem, Relaxing Durability, Quorums. Version Stamps, Business and System Transactions, Version Stamps on Multiple Nodes Textbook1: Chapter 4,5,6 RBT: L1, L2, L3</p>	08
<p>Module 3</p> <p>Map-Reduce, Basic Map-Reduce, Partitioning and Combining, Composing Map-Reduce Calculations, A Two Stage Map-Reduce Example, Incremental Map-Reduce Key-Value Databases, What Is a Key-Value Store, Key-Value Store Features, Consistency, Transactions, Query Features, Structure of Data, Scaling, Suitable Use Cases, Storing Session Information, User Profiles, Preference, Shopping Cart Data, When Not to Use, Relationships among Data, Multioperation Transactions, Query by Data, Operations by Sets Textbook1: Chapter 7,8 RBT: L1, L2, L3</p>	08
<p>Module 4</p> <p>Document Databases, What Is a Document Database?, Features, Consistency, Transactions, Availability, Query Features, Scaling, Suitable Use Cases, Event Logging, Content Management Systems, Blogging Platforms, Web Analytics or Real-Time Analytics, ECommerce Applications, When Not to Use, Complex Transactions Spanning Different Operations, Queries against Varying Aggregate Structure Textbook1: Chapter 9</p> <p>RBT: L1, L2, L3</p>	08
<p>Module 5</p> <p>Graph Databases, What Is a Graph Database?, Features, Consistency, Transactions, Availability, Query Features, Scaling, Suitable Use Cases, Connected Data, Routing, Dispatch, and Location-Based Services, Recommendation Engines, When Not to Use. Textbook1: Chapter 11 RBT: L1, L2, L3</p>	08
<p>Course Outcomes: The student will be able to :</p>	

- Define, compare and use the four types of NoSQL Databases (Document-oriented, Key Value Pairs, Column-oriented and Graph).
- Demonstrate an understanding of the detailed architecture, define objects, load data, query data and performance tune Column-oriented NoSQL databases.
- Explain the detailed architecture, define objects, load data, query data and performance tune Document-oriented NoSQL databases.

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.

Textbooks:

1. Sadalage, P. & Fowler, NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence, Pearson Addison Wesley, 2012

Reference Books:

1. Dan Sullivan, "NoSQL For Mere Mortals", 1st Edition, Pearson Education India, 2015. (ISBN13: 978-9332557338)
2. Dan McCreary and Ann Kelly, "Making Sense of NoSQL: A guide for Managers and the Rest of us", 1st Edition, Manning Publication/Dreamtech Press, 2013. (ISBN-13: 978-9351192022)
3. Kristina Chodorow, "Mongodb: The Definitive Guide- Powerful and Scalable Data Storage", 2nd Edition, O'Reilly Publications, 2013. (ISBN-13: 978-9351102694)

**MULTICORE ARCHITECTURE AND PROGRAMMING
(Effective from the academic year 2018 -2019) SEMESTER
– VII**

Course Code	18CS824	CIE Marks	40
Number of Contact Hours/Week	3:0:0	SEE Marks	60
Total Number of Contact Hours	40	Exam Hours	03
CREDITS –3			
Course Learning Objectives: This course (18CS824) will enable students to:			

<ul style="list-style-type: none"> • Define 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 -1	Contact Hours
<p>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 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.</p> <p>Textbook 1: Ch.1, 2 RBT: L1, L2, L3</p>	08
Module -2	
<p>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</p> <p>Textbook 1: Ch.3, 4 RBT: L1, L2, L3</p>	08
Module – 3	
<p>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. Textbook 1: Ch.5 RBT: L1, L2, L3</p>	08
Module-4	
<p>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 Multi-thread Execution, Data Copy-in and Copy-out, Protecting Updates of Shared Variables, Intel Task queuing Extension to OpenMP, OpenMP Library Functions,</p>	08
<p>OpenMP Environment Variables, Compilation, Debugging, performance Textbook 1: Ch.6 RBT: L1, L2, L3</p>	
Module-5	

<p>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 IA32, Data Organization for High Performance.</p> <p>Textbook 1: Ch.7 RBT: L1, L2, L3</p>	08
<p>Course Outcomes: The student will be able to :</p>	
<ul style="list-style-type: none"> • Identify the limitations of ILP and the need for multicore architectures • Define fundamental concepts of parallel programming and its design issues • Solve the issues related to multiprocessing and suggest solutions • Make out the salient features of different multicore architectures and how they exploit parallelism □ Demonstrate the role of OpenMP and programming concept 	
<p>Question Paper Pattern:</p>	
<ul style="list-style-type: none"> • 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. 	
<p>Textbooks:</p>	
<ol style="list-style-type: none"> 1. Multicore Programming , Increased Performance through Software Multi-threading by Shameem Akhter and Jason Roberts , Intel Press , 2006 	
<p>Reference Books:</p>	
<ol style="list-style-type: none"> 1. Yan Solihin, "Fundamentals of Parallel Multicore Architecture", 1st Edition, CRC Press/Taylor and Francis, 2015. 2. Gerassimos Barlas, "Multicore and GPU Programming: An Integrated Approach Paperback", 1st Edition, Morgan Kaufmann, 2014. 3. Lyla B Das, "The x86 Microprocessors: 8086 to Pentium, Multicores, Atom and the 8051 Microcontroller: Architecture, Programming and Interfacing", 2nd Edition, Pearson Education India, 2014 	