DHANRAJ BAID JAIN COLLEGE

(Autonomous)

Co-Educational Minority Institution
Owned & Managed by Tamilnadu Educational and Medical Trust
Approved by Government of Tamil Nadu & Affiliated to University of Madras
Re-Accredited by NAAC with "B+" Grade
Thoraipakkam, Chennai – 600 097.

DEPARTMENT OF COMPUTER SCIENCE

B.Sc. (Computer Science)

SYLLABUS

(Choice Based Credit System)

(Effective from the academic year 2023-24)

As per TANSCHE Syllabus

Total No. of Semesters: 6

Total No. of Credits: 141

SCIENCE PROGRAM OUTCOMES

PO1. Science: Apply the knowledge of science fundamentals to solve problems in chosen field.

PO2. Problem Analysis: Identify and review literature, thereby analyzing problems to arrive at substantiated solutions using the basic principles of science.

PO3. Science Graduate and Society: Apply reasoning to assess the societal issues like health, safety, legal and cultural to dominate the need for sustainable development.

PO4. Ethics: Apply ethical principles and commit to the professional ethics and norms.

PO5. Environment and Sustainability: To understand the impact of responsibility of pursuing the environment and demonstrate the need for sustainable development.

PO6. Individual and Team Work: Function effectively as an individual and as member or leader in diverse teams in multi – disciplinary settings.

PO7. Communication: Speak, read, write and listen clearly in person and through electronic media in English and in one Indian language, and make meaning of the world by connecting people, ideas, books, media and technology.

PO8. Project Management and Finance: Demonstrate the knowledge of the understanding of the fundamental principle of managing a project and apply the same in one ‘s own work as a member and as a leader of a team, to manage project in a multi-disciplinary environment.

PO9. Life – Long Learning: Recognize the need for and have the preparation and ability to engage in independent and lifelong learning in the broadest context of technological change

PROGRAM SPECIFIC OUTCOME

PSO1. Learning the applications of various software elements which help to identify various analysis and design methodologies.

PSO2. Demonstrate by developing computer programs in the area related to algorithm, web designing, facilitating efficient design for complex problems.

PSO3. Enables the students to be familiar with the modern- day issues, latest trends in computing and technology and create ideas and solutions to existing problems.

BACHELOR DEGREE COURSE
B.Sc. - COMPUTER SCIENCE
SEMESTER SYSTEM WITH CREDITS
(Effective from the academic year 2023-2024)

REGULATIONS

1. ELIGIBILITY FOR ADMISSION

Candidates for admission to the first year of the Degree of Bachelor of Science (Computer Science) Courses shall be required to have passed the Higher Secondary Examinations, having Mathematics/Statistics/ Computer Science as one of the subject, conducted by the Government of Tamil Nadu or an Examination accepted as equivalent thereof by the Syndicate of the University of Madras.

2. ELIGIBILITY FOR THE AWARD OF DEGREE

A Candidate shall be eligible for the award of the degree only if he/she has undergone the prescribed course of study in our College which is affiliated to the University of Madras for a period of not less than three academic years, passed the examinations of all the Six Semesters prescribed earning 140 credits and fulfilled such conditions as have been prescribed therefor.

3. DURATION

a) Each academic year shall be divided into two semesters. The first academic year shall comprise the first and second semesters, the second academic year comprise of the third and fourth semesters and the third academic year shall comprise the fifth and sixth semesters respectively.
b) The odd semesters shall consist of the period from June to November of each year and the even semesters from December to April of each year. There shall be not less than 90 working days for each semester.

4. COURSE OF STUDY

The Main Subject of study for Bachelor Degree shall consist of the following

PART-I: Tamil or any other Modern (Indian or Foreign) or Classical Languages.
PART-II: English
PART-III: CORE COURSES Consisting of (a) Core Course (b) Elective Course (Generic / Discipline Specific)
PART-IV: Skill Enhancement Course (SEC), Foundation Course (FC), Environmental Studies, Value Education, Internship/ Industrial Training, Professional Competency Skill Enhancement Course.
PART- V: Extension Activity

5. SCHEME OF EXAMINATION

Scheme of Examination shall be enclosed in APPENDIX - I

6. REQUIREMENTS FOR PROCEEDING TO SUBSEQUENT SEMESTER

I. Candidates shall register their names for the First Semester Examination after the admission in the UG Courses.
II. Candidates shall be permitted to proceed from the First Semester up to final semester irrespective of their failure in any of the semester examinations subject to the condition that the candidates should register for all the arrear subjects of earlier semesters along with current (subsequent) Semester Subjects.
III. Candidates shall be eligible to go to subsequent semester, only if they earn, sufficient attendance as prescribed therefore by the syndicate from time to time, provided in case of a candidate earning less than 50% of attendance in any one of the Semesters due to any extraordinary circumstance such as medical grounds, such candidates who shall produce Medical Certificate issued by the Authorised Medical Attendant (AMA), duly certified by the Principal of the College, shall be permitted to proceed to the next semester and to complete the Course of Study. Such candidates shall have to repeat the missed semester by rejoining after completion of Final Semester of the course, after paying the fee for the break of study as prescribed by the University from time to time.

7. PASSING MINIMUM

A candidate shall be declared to have passed in each paper/practical of the Main Subject of study wherever prescribed, if he/she secures NOT LESS THAN 40% of the marks prescribed for the examination. He/she shall be declared to have passed the whole examination, if he/she passes in all the papers and practicals wherever prescribed as per the scheme of examinations earning 140 CREDITS. He/she shall also fulfill the compulsory extension services prescribed earning a minimum of 1 credit out of maximum of 2 credits to qualify for the degree.

8. CLASSFICATION OF SUCCESSFUL CANDIDATES

i) PART I: LANGUAGE OTHER THAN ENGLISH: Successful candidates passing the examinations for the Language and securing the marks (1) 60 percent and above and (2) 50 percent and above but below 60 percent in the aggregate shall be declared to have passed the examination in the FIRST and SECOND Class, respectively. All other successful candidates shall be declared to have passed the examination in the THIRD Class.

ii) PART II: ENGLISH: Successful candidates passing the examinations for English and securing the marks (1) 60 percent and above and (2) 50 percent and above but below 60 percent in the aggregate shall be declared to have passed the examination in the FIRST and SECOND class, respectively. All other successful candidates shall be declared to have passed the examination in the THIRD class.

iii) PART III: CORE COURSE: Consisting of (a) Core Course (b) Elective Course (Generic / Discipline Specific)

Successful candidates passing the examinations for Core Courses together and securing the marks 1)60 percent and above (2) 50 percent and above but below 60 percent in the aggregate of the marks prescribed for the Core Courses together shall be declared to have passed the examination in the FIRST and SECOND class respectively. All other successful candidates shall be declared to have passed the examinations in the THIRD class.

iv) PART IV: Skill Enhancement Course (SEC), Foundation Course (FC), Environmental Studies, Value Education, Internship/ Industrial Training, Professional Competency Skill Enhancement Course.

v) PART V: COMPULSORY EXTENSION SERVICE:

Successful Candidates earning a minimum of 1 credit or a maximum of 2 credits SHALL NOT BE taken into consideration for Classification/Ranking/Distinction.

9. RANKING

Candidates who pass all the examinations prescribed for the course in the FIRST APPEARANCE ITSELF ALONE are eligible for Classification/Ranking/Distinction.

Provided in the case of Candidates who pass all the examinations prescribed for the course with a break in the First Appearance due to the reasons as furnished in the Regulations.7. (iii) supra are only eligible for Classification / Distinction.

PATTERN OF QUESTION PAPER

Maximum Marks: - 75 Marks **Time: - 3 Hours**

Part – A (50 Words)

To answer any TEN Questions out of Twelve Questions 10x2=20 Marks

Part – B (200 Words)

To answer any FIVE Questions out of Eight Questions 5 x 5=25 Marks

Part – C (500 Words)

To answer Any THREE Questions out of Five Questions 3x10=30 Marks

QUESTION PAPER FOR PRACTICALS

The external examiner will prepare a question paper on the Spot with the help of the Question Bank Supplied by the Controller's Office.

B.Sc. Computer Science

First Year

Semester-I

Part	List of Courses	Credit	Hours per week (L/T/P)
Part-I	Language	3	5
Part-II	English	3	5
Part-III	CC1-Object Oriented Programming with C++	5	5
	CC2- Practical – Object Oriented Programming Lab	5	5
	Elective Course - (Generic / Discipline Specific) - EC1 - Statistics - I	3	5
Part-IV	Skill Enhancement Course SEC-1 (Non Major Elective)- Office Automation Lab	2	3
	Foundation Course FC- Problem Solving Techniques	2	2
		23	**30**

Semester-II

Part	List of Courses	Credit	Hours per week (L/T/P)
Part-I	Language	3	5
Part-II	English	3	5
Part-III	CC3-Data Structure and Algorithm	5	5
	CC4-Data Structure and Algorithm Lab	5	5
	Elective Course - (Generic / Discipline Specific) - EC2 - Resource Management Techniques	3	5
Part-IV	Skill Enhancement Course -SEC-2 (Non Major Elective) – Quantitative Aptitude	2	3
	Skill Enhancement Course -SEC-3 (Discipline Specific / Generic)- Advanced Excel Lab	2	2
		23	**30**

Second Year

Semester-III

Part	List of Courses	Credit	Hours per week(L/T/P)
Part-I	Language	3	5
Part-II	English	3	5
Part-III	CC5- Python Programming	5	5
	CC6 - Practical: Python Programming Lab	5	5
	Elective Course 3 (Generic / Discipline Specific) - EC3 - Mathematics - I	3	5
Part-IV	Skill Enhancement Course -SEC-4 (Entrepreneurial Based) – Web Application Development	2	2
	Skill Enhancement Course -SEC-5 (Discipline Specific/ Generic) – Web Application Development Lab	2	2
	Environmental Studies	-	1
		23	**30**

Semester-IV

Part	List of Courses	Credit	Hours per week (L/T/P)
Part-I	Language	3	5
Part-II	English	3	5
Part-III	CC7 - Industry Module –Java Programming	5	5
	CC8 - Practical: Java Programming Lab	5	5
	Elective Course - EC4 (Generic / Discipline Specific) – Mathematics - II	3	5
Part-IV	Skill Enhancement Course – SEC-6 - PHP Programming	2	2
	Skill Enhancement Course - SEC-7 – Practical- PHP Programming Lab	2	2
	Environmental Studies	2	1
		25	**30**

Third Year

Semester-V

Part	List of Courses	Credit	Hours per week (L/T/P)
Part-III	CC9 - Operating Systems	4	5
	CC10 - Database Management System	4	5
	CC11 - Practical - Operating Systems Lab	4	5
	CC12 – Practical-Database Management System Lab	4	5
	Elective Course – DSE-I (Generic / Discipline Specific) – Computer Network	3	5
	Elective Course – DSE-II (Generic / Discipline Specific) – Introduction to Data Science	3	4
Part-IV	Value Education	2	1
	Internship / Industrial Training (Summer vacation at the end of IV semester activity)	2	
		26	**30**

Semester-VI

Part	List of Courses	Credit	Hours per week (L/T/P)
Part-III	CC13 -.NET Programming	4	5
	CC14 - Practical: .NET Programming	4	5
	CC15 - Mini Project	4	5
	Elective Course – DSE-III (Generic / Discipline Specific) – Cyber Security	3	5
	Elective Course – DSE-IV (Generic / Discipline Specific) – Software Engineering	3	5
Part-IV	Professional Competency Skill Enhancement Course PCSEC – Cloud Computing	2	5
Part -V	Extension Activity	1	
		21	**30**

Total Credits: 141

First Year (Semester I)

Title of the Course/ Paper	Subject Name	Category	L	T	P	S	Credits	Inst. Hours	Marks		
									CIA	External	Total
Core Course I	**OBJECT ORIENTED PROGRAMMING WITH C++**	Core	Y	-	-	-	5	5	25	75	100

Course Objective	
C1	Describe the procedural and object oriented paradigm with concepts of streams, classes,functions, data and objects
C2	Understand dynamic memory management techniques using pointers, constructors,destructors, etc
C3	Describe the concept of function overloading, operator overloading, virtual functions and polymorphism
C4	Classify inheritance with the understanding of early and late binding, usage of exception handling, generic programming
C5	Demonstrate the use of various OOPs concepts with the help of programs

UNIT	**Details**	**No. of Hours**
I	Introduction to C++ - key concepts of Object-Oriented Programming – Advantages – Object Oriented Languages – I/O in C++ - C++ Declarations. Control Structures: - Decision Making and Statements: If . else, jump, goto, break, continue, Switch case statements - Loops in C++ :for, while, do - functions in C++ - inline functions – Function Overloading.	15
II	Classes and Objects: Declaring Objects – Defining Member Functions – Static Member variables and functions – array of objects –friend functions – Overloading member functions – Bit fields and classes – Constructor and destructor with static members.	15
III	Operator Overloading: Overloading unary, binary operators – Overloading Friend functions –type conversion – Inheritance: Types of Inheritance – Single, Multilevel, Multiple, Hierarchal, Hybrid, Multi path inheritance – Virtual base Classes – Abstract Classes.	15

IV	Pointers – Declaration – Pointer to Class , Object – this pointer – Pointers to derived classes and Base classes – Arrays – Characteristics – array of classes – Memory models – new and delete operators – dynamic object – Binding, Polymorphism and Virtual Functions.	15
V	Files – File stream classes – file modes – Sequential Read / Write operations – Binary and ASCII Files – Random Access Operation – Templates – Exception Handling - String – Declaring and Initializing string objects – String Attributes – Miscellaneous functions .	15
	Total	**75**

Course Outcomes		**Programme Outcome**
CO	Upon completion of the course the students would be able to:	
1	Remember the program structure of C with its syntax and semantics	PO1,PO6
2	Understand the programming principles in C (data types, operators, branching and looping, arrays, functions, structures, pointers and files)	PO2
3	Apply the programming principles learnt in real-time problems	PO4 ,PO7
4	Analyze the various methods of solving a problem and choose the best method	PO6
5	Code, debug and test the programs with appropriate test cases	PO7,PO8

Text Book	
1	E. Balagurusamy, “Object-Oriented Programming with C++”, TMH 2013, 7th Edition.
Reference Books	
1.	Ashok N Kamthane, “Object-Oriented Programming with ANSI and Turbo C++”, Pearson Education 2003.
2.	Maria Litvin& Gray Litvin, “C++ for you”, Vikas publication 2002.
Web Resources	
	https://alison.com/course/introduction-to-c-plus-plus-programming

Title of the Course/ Paper	Subject Name	Category	L	T	P	S	Credits	Inst. Hours	Marks		
									CIA	External	Total
Core Course II	OBJECT ORIENTED PROGRAMMING LAB	Core	-	-	Y	-	5	5	25	75	100

Course Objective	
C1	Describe the procedural and object oriented paradigm with concepts of streams, classes, functions, data and objects
C2	Understand dynamic memory management techniques using pointers, constructors, destructors, etc
C3	Describe the concept of function overloading, operator overloading, virtual functions and polymorphism
C4	Classify inheritance with the understanding of early and late binding, usage of exception handling, generic programming
C5	Demonstrate the use of various OOPs concepts with the help of programs

S.No	Details	No. of Hours
1	Write a C++ program to demonstrate function overloading, Default Arguments and Inline function.	
2	Write a C++ program to demonstrate Class and Objects	
3	Write a C++ program to demonstrate the concept of Passing Objects to Functions	
4	Write a C++ program to demonstrate the Friend Functions.	
5	Write a C++ program to demonstrate the concept of Passing Objects to Functions	
6	Write a C++ program to demonstrate Constructor and Destructor	
7	Write a C++ program to demonstrate Unary Operator Overloading	
8	Write a C++ program to demonstrate Binary Operator Overloading	

9	Write a C++ program to demonstrate: • Single Inheritance • Multilevel Inheritance • Multiple Inheritance • Hierarchical Inheritance • Hybrid Inheritance	
10	Write a C++ program to demonstrate Virtual Functions.	
11	Write a C++ program to manipulate a Text File.	
12	Write a C++ program to perform Sequential I/O Operations on a file.	
13	Write a C++ program to find the Biggest Number using Command Line Arguments	
14	Write a C++ program to demonstrate Class Template	
15	Write a C++ program to demonstrate Function Template.	
16	Write a C++ program to demonstrate Exception Handling.	

Course Outcomes		**Programme Outcome**
CO	Upon completion of the course the students would be able to:	
1	Remember the program structure of C with its syntax and semantics	PO1,PO6
2	Understand the programming principles in C (data types, operators, branching and looping, arrays, functions, structures, pointers and files)	PO2
3	Apply the programming principles learnt in real-time problems	PO4 ,PO7
4	Analyze the various methods of solving a problem and choose the best method	PO6
5	Code, debug and test the programs with appropriate test cases	PO7,PO8

Text Book	
1	E. Balagurusamy, “Object-Oriented Programming with C++”, TMH 2013, 7th Edition.
Reference Books	

1.	Ashok N Kamthane, "Object-Oriented Programming with ANSI and Turbo C++", Pearson Education 2003.
2.	Maria Litvin& Gray Litvin, "C++ for you", Vikas publication 2002.
Web Resources	
1.	https://alison.com/course/introduction-to-c-plus-plus-programming

Elective Course (EC1) : Statistics-I	
Lecture Hours: 5 per week	**Credits: 3**

Learning Objectives: (for teachers: what they have to do in the class/lab/field)

Course Outcomes: (for students: To know what they are going to learn)

1. Know the uses of statistics in society
2. Organize, manage and present data
3. Analyze the statistical data graphically using frequency distribution and cumulative frequency distribution.
4. Analyze statistical data using measures of central tendency, dispersion and location.
5. To understand correlation between continuous variables and association between categorical variables.

Units	Contents
I	Methods of collection: Complete enumeration – Sample Survey - Primary data - Secondary data sources - Types of variables. Nominal, ordinal and scale data. Presentation of Data: Presentation of data by tables - construction of tables (Univariate and Bivariate) – frequency table and contingency table.
II	Diagrammatic presentation: Line diagram, Bar diagrams: Simple, multiple, subdivided and Percentage-Pie chart, comparative pie chart - Graphical representation of a frequency distribution by histogram and frequency polygon and Ogives.
III	Analysis of Data (Univariate): Measures of central tendency: Arithmetic mean-Median and Mode choice of an average-characteristic of a good average
IV	Measures of dispersion: Range-Quartile deviation-mean deviation - standard deviation - relative measures of dispersion - Coefficient of Variance
V	Analysis of Data (Bivariate): Correlation- Scatter plot-coefficient of correlation-Pearson's Correlation Coefficient, Spearmen's rank correlation coefficient-correlation coefficient for bivariate frequency table.

Suggested Readings:

Books for study:

1. Gupta, S.C and Kapoor, V. K (2002), *Fundamentals of Mathematical Statistics*, Sultan Chand and Sons, New Delhi.
2. Goon A.M., Gupta M.K. and Dasgupta B. (2002): *Fundamentals of Statistics*, Vol. I & II, 8th Edn. The World Press, Kolkata.
3. Irwin Miller, Marylees Miller (2006): *John E. Freund's Mathematical Statistics with Applications*, (7th Edn.), Prentice Hall International INC.
4. Mood, A.M. Graybill, F.A. and Boes, D.C. (2007): *Introduction to the Theory of Statistics*, 3rd Edn., (Reprint), Tata McGraw-Hill Pub. Co. Ltd

Books for reference:

1. Saxena H.C.: *Elementary Statistics*. S. Chand & Co., 2009.

Course Code: SEC-1	**Office Automation Lab**		**Credits: 2**
Lecture Hours: (L) **per week: -**	**Tutorial Hours:** **(T) per week**	**Lab Practice** **Hours: 3 per week**	**Total: (L+T+P)** **per week: 3**
Course Category : Skill Enhancement	**Year & Semester: I Year I Semester**	**Admission Year:**	
Pre-requisite	Basic skills in Computer operations		

Learning Objectives: (for teachers: what they have to do in the class/lab/field)

- The major objective in introducing the Computer Skills course is to impart training for students in Microsoft Office which has different components like MS Word, MS Excel and Power point.
- The course is highly practice oriented rather than regular class room teaching.
- To acquire knowledge on editor, spread sheet and presentation software.

Course Outcomes: (for students: To know what they are going to learn)

CO1: Understand the basics of computer systems and its components.

CO2: Understand and apply the basic concepts of a word processing package.

CO3: Understand and apply the basic concepts of electronic spreadsheet software.

CO4: Understand and apply the basic concepts of database management system.

CO5: Understand and create a presentation using PowerPoint tool.

Recap: (not for examination) Motivation/previous lecture/ relevant portions required for the course) [This is done during 2 Tutorial hours)

S.no	Contents	Required Hours
I	LIST OF PROGRAMS: I. WORD PROCESSING 1. Text manipulation Change the font size and type Aligning and justification of text Underlining the Text Indenting the Text i. Prepare a Bio-Data ii. Prepare a letter 2. Usage of Numbering, Bullets, Footers and Headers Usage of Spell checks and Find and Replace i. Prepare a document in newspaper format ii. Prepare a document with bullets and footers and headers. 3.Tables and Manipulations Creations, Insertion, Deletion (Columns & Rows) and usage of Auto Format i. Create a mark sheet using table and find out the total marks. ii. Create a calendar and Auto format it 4.Picture Insertion and alignment i. Prepare a greeting card ii. Prepare a handout	

	5. Creation of documents using templates Creation of Templates i. Prepare a letter using any template ii. Prepare two data using various kinds of templates 6. Mail Merge concepts i. Prepare a business letter for more than one company using mail merge ii. Prepare an invitation to be sent to specific addresses in the data source. 7. Copying text and pictures from Excel i. Draw a chart in Excel and paste it on word ii. Import a picture from Excel and edit the picture. II. MS-EXCEL 1. Prepare a Mark List for students (use Conditional Formatting). 2. Arrange data in Ascending and Descending order. 3. Pay bill Preparation. 4. Prepare a Calendar - Auto formatting 5. EB bill Preparation. 6. Creating a chart. 7. Prepare an Inventory bill for a company (use Freeze Panes, Track Changes). 8. Insertion, Deletion, Merging and Formatting of cells III-MS POWERPOINT 1. Design presentation slides for a product of your choice. The slides must include name, brand name, type of product, characteristics, special features, price, special offer etc. Add voice if possible to explain the features of the product. The presentation should work in manual mode. (Apply Animation schemes and Slide Transition) 2. Design slides for the headlines News of a popular TV Channel. The Presentation Should contain the following transactions: Top down, Bottom up, Zoom in and Zoom out. The presentation should work in custom mode. 3. Animate a Smile Face (Cry, Normal, Smile).	

Learning Resources:

- **Recommended Texts**
  1. Peter Norton, "Introduction to Computers" –Tata McGraw-Hill.
- **Reference Books**
  1. Jennifer Ackerman Kettel, Guy Hat-Davis, Curt Simmons, "Microsoft 2003", Tata McGraw- Hill.
- **Web resources:** Web content from NDL / SWAYAM or open source web resources

Course Code: FC	**Problem Solving Techniques**		**Credits: 2**
Lecture Hours: (L) per week: 2	**Tutorial Hours: (T) per week**	**Lab Practice Hours: (P)per week**	**Total: (L+T+P) per week: 2**
Course Category : Foundation	**Year & Semester: I Year I Semester**		**Admission Year:**
Pre-requisite	Basic of Problem-solving skills		

Learning Objectives:

- To understand the importance of algorithms and programs, and to know of the basic problem solving strategies.
- To learn efficient strategies and algorithms to solve standard problems, thus laying a firm foundation for designing algorithmic solutions to problems.

Course Outcomes: (for students: To know what they are going to learn)

CO1: Understand the systematic approach to problem solving.

CO2: Know the approach and algorithms to solve specific fundamental problems.

CO3: Understand the efficient approach to solve specific factoring-related problems.

CO4: Understand the efficient array-related techniques to solve specific problems.

CO5: Understand the efficient methods to solve specific problems related to text processing. Understand how recursion works.

Recap: (not for examination) Motivation/previous lecture/ relevant portions required for the course) [This is done during 2 Tutorial hours)

Units	**Contents**	**Required Hours**

I	**Introduction:** Notion of algorithms and programs – Requirements for solving problems by computer – The problem-solving aspect: Problem definition phase, Getting started on a problem, The use of specific examples, Similarities among problems, Working backwards from the solution – General problem-solving strategies - Problem solving using top-down design – Implementation of algorithms – The concept of Recursion.	**06**
II	**Fundamental Algorithms**: Exchanging the values of two variables – Counting - Summation of a set of numbers - Factorial computation - Sine function computation - Fibonacci Series generation - Reversing the digits of an integer – Base Conversion.	**06**
III	**Factoring Methods**: Finding the square root of a number – The smallest divisor of an integer – Greatest common divisor of two integers - Generating prime numbers – Computing the prime factors of an integer – Generation of pseudo-random numbers - Raising a number to a large power – Computing the *n*th Fibonacci number.	**06**
IV	**Array Techniques**: Array order reversal – Array counting or histogramming – Finding the maximum number in a set - Removal of duplicates from an ordered array - Partitioning an array – Finding the k^{th} smallest element – Longest monotone subsequence.	**06**
V	**Text Processing and Pattern Searching**: Text line length adjustment – Left and right justification of text – Keyword searching in text – Text line editing – Linear pattern search. **Recursive algorithms**: Towers of Hanoi – Permutation generation.	**06**

Extended Professional Component (is a part of internal component only, Not to be included in the External Examination question paper)	Questions related to the above topics, from various competitive examinations UPSC / TRB / NET / UGC – CSIR / GATE / TNPSC / others to be solved (To be discussed during the Tutorial hour)	
Skills acquired from the course	Knowledge, Problem Solving, Analytical ability, Professional Competency, Professional Communication and Transferrable Skill	

Learning Resources:

- **Recommended Texts**

1. R. G. Dromey, *How to Solve it by Computer*, Pearson India, 2007.

- **Reference Books**

1. George Polya, Jeremy Kilpatrick, *The Stanford Mathematics Problem Book: With Hints and Solutions*, Dover Publications, 2009 (Kindle Edition 2013).
2. Greg W. Scragg, *Problem Solving with Computers*, Jones & Bartlett 1st edition, 1996.

- **Web resources**

First Year

(Semester II)

Course Code: CC3	**Data Structures and Algorithms**		**Credits: 5**
Lecture Hours: (L) per week: 5	**Tutorial Hours: (T) per week**	**Lab Practice Hours: (P)per week**	**Total: (L+T+P) per week: 5**
Course Category : Core	**Year & Semester: I Year II Semester**	**Admission Year:**	
Pre-requisite	Basic knowledge in data and representations		
Links to other Courses			

Learning Objectives: (for teachers: what they have to do in the class/lab/field)

- To impart the basic concepts of data structures and algorithms.
- To acquaint the student with the basics of the various data structures and make the students knowledgeable in the area of data structures.
- This course also gives insight into the various algorithm design techniques

Course Outcomes: (for students: To know what they are going to learn)

CO1: To introduce the concepts of Data structures and to understand simple linear data structures.

CO2: Learn the basics of stack data structure, its implementation and application

CO3: Use the appropriate data structure in context of solution of given problem and demonstrate a familiarity with major data structures.

CO4: To introduce the basic concepts of algorithms

CO5: To give clear idea on algorithmic design paradigms like Dynamic Programming, Backtracking, Branch and Bound

Recap: (not for examination) Motivation/previous lecture/ relevant portions required for the course) [This is done during 2 Tutorial hours)

Units	**Contents**	**Required Hours**
I	**INTRODUCTION TO DATA STRUCTURES:**	**15**

	• Data Structures: Definition- Time & Space Complexity, • Arrays: Representation of arrays, Applications of arrays, sparse matrix and its representation, • Linear list: Singly linked list implementation, insertion, deletion and searching operations on linear list • Circular linked list: implementation, Double linked list implementation, insertion, deletion and searching operations. Applications of linked lists- Dynamic Storage management.	
II	**STACKS:** • Operations, array and linked representations of stack, • stack applications, infix to postfix conversion, postfix expression evaluation, recursion implementation	**15**
III	**QUEUES, TREES & GRAPHS:** • **Queues**: operations on queues, array and linked representations. • **Circular Queue:** operations, applications of queues. • **Trees:** Definitions and Concepts- Representation of binary tree, Binary tree traversals (Inorder, Postorder, Preorder), • Binary search trees • **Graphs** : Representation of Graphs- Types of graphs -Breadth first traversal – Depth first traversal- - Applications of graphs –	**15**
IV	**INTRODUCTION TO ALGORITHMS:** • **INTRODUCTION**: Definition of Algorithms- Overview and importance of algorithms- pseudocode	**15**

	conventions, Asymptotic notations, practical complexities. ● **Divide-and-Conquer:** General Method – Binary Search- Quick Sort- Merge Sort. ● **Greedy Method:** General method- Knapsack problem- Tree vertex splitting- Job sequencing with deadlines	
V	**DYNAMIC PROGRAMMING, BACKTRACKING & BRANCH & BOUND** ● **Dynamic programming**: General method, Multistage Graphs, All pairs shortest path, Single source shortest path. ● **Backtracking:** General method, 8 Queens, Graph coloring, Hamiltonian cycle. ● **Branch & Bound**: General method, Travelling salesperson problem.	**15**
Extended Professional Component (is a part of internal component only, Not to be included in the External Examination question paper)	Questions related to the above topics, from various competitive examinations UPSC / TRB / NET / UGC – CSIR / GATE / TNPSC / others to be solved (To be discussed during the Tutorial hour)	

Skills acquired from the course	Knowledge, Problem Solving, Analytical ability, Professional Competency, Professional Communication and Transferrable Skill	

Learning Resources:

- **Recommended Texts**

  1. Ellis Horowitz , Sartaj Sahni, Susan Anderson Freed, Second Edition , "Fundamentals of Data in C", Universities Press

  2. E. Horowitz, S. Sahni and S. Rajasekaran, Second Edition ,"Fundamentals of Computer Algorithms " Universities Press

- **Reference Books**

  1. Seymour Lipschutz ,"Data Structures with C", First Edition, Schaum's outline series in computers, Tata McGraw Hill.
  2. R.Krishnamoorthy and G.Indirani Kumaravel, Data Structures using C, Tata McGrawHill – 2008.
  3. A.K.Sharma, Data Structures using C , Pearson Education India,2011.
  4. G. Brassard and P. Bratley, "Fundamentals of Algorithms", PHI, New Delhi, 1997.
  5. A.V. Aho, J.E. Hopcroft, J.D. Ullmann,, "The design and analysis of Computer
  6. Algorithms", Addison Wesley, Boston, 1974
  7. Thomas H. Cormen, C.E. Leiserson, R L.Rivest and C. Stein, Introduction to Algorithms, Third edition, MIT Press, 2009
  8. Sanjoy Dasgupta, C.Papadimitriou and U.Vazirani , Algorithms , Tata McGraw-Hill, 2008.

- **Web resources:** Web resources from NDL Library, E-content from open source libraries

<table>
<tr><td colspan="2">Course Code: CC4</td><td colspan="2">Data Structures and Algorithms Lab</td><td>Credits: 5</td></tr>
<tr><td>Lecture Hours: (L) per week</td><td>Tutorial Hours : (T) per week</td><td>Lab Practice Hours: 5 per week</td><td>Total: (L+T+P) per week: 5</td></tr>
<tr><td>Course Category : Core</td><td colspan="2">Year & Semester: I Year II Semester</td><td>Admission Year:</td></tr>
<tr><td>Pre-requisite</td><td colspan="3">Basic skills in problem solving</td></tr>
<tr><td colspan="4">Learning Objectives: (for teachers: what they have to do in the class/lab/field)
• To understand and implement basic data structures using C
• To apply linear and non-linear data structures in problem solving.
• To learn to implement functions and recursive functions by means of data structures
• To implement searching and sorting algorithms</td></tr>
<tr><td colspan="4">Course Outcomes: (for students: To know what they are going to learn)
CO1: Implement data structures using C
CO2: Implement various types of linked lists and their applications
CO3: Implement Tree Traversals
CO4: Implement various algorithms in C
CO5: Implement different sorting and searching algorithms</td></tr>
<tr><td colspan="4">Recap: (not for examination) Motivation/previous lecture/ relevant portions required for the course) [This is done during 2 Tutorial hours)</td></tr>
<tr><td></td><td colspan="2">List of Exercises:</td><td>Required Hours</td></tr>
<tr><td></td><td colspan="2">Implement the following exercises using C Programming language:

1. Array implementation of stacks
2. Array implementation of Queues
3. Linked list implementation of stacks
4. Linked list implementation of Queues
5. Binary Tree Traversals (Inorder, Preorder, Postorder)
6. Implementation of Linear search and binary search
7. Implementation Insertion sort, Quick sort and Merge Sort</td><td></td></tr>
</table>

	8. Implementation of Depth-First Search & Breadth-First Search of Graphs. 9. Finding all pairs of Shortest Path of a Graph. 10. Finding single source shortest path of a Graph.	
Extended Professional Component (is a part of internal component only, Not to be included in the External Examination question paper)	Questions related to the above topics, from various competitive examinations UPSC / TRB / NET / UGC – CSIR / GATE / TNPSC / others to be solved (To be discussed during the Tutorial hour)	
Skills acquired from the course	Knowledge, Problem Solving, Analytical ability, Professional Competency, Professional Communication and Transferrable Skill	

Learning Resources:

Learning Resources:

- **Recommended Texts**

  1. Ellis Horowitz , Sartaj Sahni, Susan Anderson Freed, Second Edition , "Fundamentals of Data in C", Universities Press

  2. E. Horowitz, S. Sahni and S. Rajasekaran, Second Edition ,"Fundamentals of Computer Algorithms " Universities Press

- **Reference Books**

1. Seymour Lipschutz ,"Data Structures with C", First Edition, Schaum's outline series in computers, Tata McGraw Hill.

2. R.Krishnamoorthy and G.Indirani Kumaravel, Data Structures using C, Tata McGrawHill – 2008.

3. A.K.Sharma, Data Structures using C , Pearson Education India,2011.

4. G. Brassard and P. Bratley, "Fundamentals of Algorithms", PHI, New Delhi, 1997.

5. A.V. Aho, J.E. Hopcroft, J.D. Ullmann,, "The design and analysis of Computer

6. Algorithms", Addison Wesley, Boston, 1974

7. Thomas H. Cormen, C.E. Leiserson, R L.Rivest and C. Stein, Introduction to Algorithms, Third edition, MIT Press, 2009

8. Sanjoy Dasgupta, C.Papadimitriou and U.Vazirani , Algorithms , Tata McGraw-Hill, 2008.

- **Web resources:** Web resources from NDL Library, E-content from open source libraries

Course Code: EC2	**Resource Management Techniques**		Credits: 3
Lecture Hours: (L) per week: 5	Tutorial Hours : (T) per week	Lab Practice Hours: (P)per week	Total: (L+T+P) per week: 5
Course Category : Elective	Year & Semester: I Year II Semester		Admission Year:
Pre-requisite	Basic Knowledge on LPP		
Learning Objectives: (for teachers: what they have to do in the class/lab/field) To understand the mathematical concepts like LPP, graphical solutions			

Course Outcomes: (for students: To know what they are going to learn)

CO1: To gain knowledge on LPP

CO2: Able to understand different mathematical models

CO3: To get an idea on game theory

CO4: Understanding the different form of sequencing problem

CO5: Able to understand Relations and its applications of transportation and assignment model

Recap: (not for examination) Motivation/previous lecture/ relevant portions required for the course) [This is done during 2 Tutorial hours)

Units	Contents	Required Hours
I	Linear Programming: Formulation of different types of linear programming – meaning of linear programming-Canonical and standard form of linear programming-Methods for solving LPP– Mathematical formulation of L.P.P – Graphical method -Simplex methods. Simple problems.	
II	Transportation Model – Mathematical formulation of a transportation problem – Basics of transportation problem -Methods for finding initial basic feasible solution – Formulation and solving methods-North west corner rule , Least cost entry method and Vogel's Approximation method - Balanced and Unbalanced transportation problem	

III	*Assignment problem – Mathematical formulation of an assignment problem – Assignment Algorithm – Balanced and Unbalanced assignment problem –Simple problems.*	
IV	Sequencing problems – Processing with N jobs through two machines- Processing with N jobs through three machines-simple problems	
V	Game theory –Two person zero Sum game - Maxmin – Minmax principle – Saddle point and value of the Game – Game without saddle point, Mixed strategies – Dominance property Graphical method for 2xn and mx2 games	
Extended Professional Component (is a part of internal component only, Not to Be Included in The External Examination Question paper)	Questions related to the above topics, from various competitive examinations UPSC / TRB / NET / UGC – CSIR / GATE / TNPSC / others to be solved (To be discussed during the Tutorial hour)	

Skills acquired from the Course	Knowledge, Problem Solving, Analytical ability, Professional Competency, Professional Communication andTransferrable Skill	

Contents and treatments as in

Sundaresan V and K.S.Ganapathy Subramanian, Contents and Treatment in ResourceManagement Techniques, 4 th Ed., A.R.Publications, Chennai.

Reference Books:

1. Kanti Swarup, P.K.Gupta, Man Mohan, Operations Research, 15 th Ed., Sultan Chand& Sons, New Delhi, 2010

2. Prem Kumar Gupta, D.S. Hira, Web resources: Web resources from NDL Library, E-content from open-sourcelibraries

Course Code: SEC-2	**Quantitative Aptitude**		**Credits: 2**
Lecture Hours: (L) per week: 3	**Tutorial Hours: (T) per week**	**Lab Practice Hours: (P)per week**	**Total: (L+T+P) per week: 3**
Course Category :Skill Enhancement	**Year & Semester: I Year II Semester**		**Admission Year:**
Pre-requisite	Basic knowledge in numerical ability		

Learning Objectives: (for teachers: what they have to do in the class/lab/field)

- To improve the quantitative skills of the students
- To prepare the students for various competitive exams

Course Outcomes: (for students: To know what they are going to learn)

CO1: To gain knowledge on LCM and HCF and its related problems

CO2: To get an idea of age, profit and loss related problem solving.

CO3: Able to understand time series simple and compound interests

CO4: Understanding the problem related to probability, and series

CO5: Able to understand graphs, charts

Recap: (not for examination) Motivation/previous lecture/ relevant portions required for the course) [This is done during 2 Tutorial hours)

Units	Contents	Required Hours
I	Numbers - HCF and LCM of numbers - Decimal fractions - Simplification - Square roots and cube roots - Average - problems on Numbers	**06**
II	Problems on Ages - Surds and Indices - percentage - profits and loss - ratio and proportion - partnership - Chain rule.	**06**
III	Time and work - pipes and cisterns - Time and Distance - problems on trains - Boats and streams - simple interest - compound interest - Logarithms - Area - Volume and surface area - races and Games of skill.	**06**
IV	Permutation and combination - probability - True Discount - Bankers Discount - Height and Distances - Odd man out & Series.	**06**

V	Calendar - Clocks - stocks and shares - Data representation - Tabulation - Bar Graphs - Pie charts - Line graphs	**06**
Extended Professional Component (is a part of internal component only, Not to be included in the External Examination question paper)	Questions related to the above topics, from various competitive examinations UPSC / TRB / NET / UGC – CSIR / GATE / TNPSC / others to be solved (To be discussed during the Tutorial hour)	
Skills acquired from the course	Knowledge, Problem Solving, Analytical ability, Professional Competency, Professional Communication and Transferrable Skill	

Learning Resources:

- **Recommended Texts**

1. . "Quantitative Aptitude", R.S. AGGARWAL., S. Chand & Company Ltd.,

- **Web resources: Authentic** Web resources related to Competitive examinations

<table>
<tr><td colspan="2">Course Code: SEC-3</td><td colspan="2">Advanced Excel Lab</td><td>Credits: 2</td></tr>
<tr><td>Lecture Hours: (L) per week: 2</td><td>Tutorial Hours: (T) per week</td><td colspan="2">Lab Practice Hours: (P)per week</td><td>Total: (L+T+P) per week: 2</td></tr>
<tr><td>Course Category : Skill Enhancement</td><td colspan="2">Year & Semester: I Year II Semester</td><td colspan="2">Admission Year:</td></tr>
<tr><td>Pre-requisite</td><td colspan="4">Basic knowledge in office automation / Excel</td></tr>
<tr><td colspan="5">Learning Objectives: (for teachers: what they have to do in the class/lab/field)
The objective of this course is to help the students learn the advanced features of Excel, to summarize, analyze, explore, and present visualizations of data in the form of charts, graphs.</td></tr>
<tr><td colspan="5">Course Outcomes: (for students: To know what they are going to learn)
CO1: Handle large amounts of data
CO2: Aggregate numeric data and summarize into categories and subcategories
CO3: Filtering, sorting, and grouping data or subsets of data
CO4: Create pivot tables to consolidate data from multiple files
CO5: Presenting data in the form of charts and graphs</td></tr>
<tr><td colspan="5">Recap: (not for examination) Motivation/previous lecture/ relevant portions required for the
course) [This is done during 2 Tutorial hours)</td></tr>
<tr><td>Units</td><td colspan="3">Contents</td><td>Required Hours</td></tr>
</table>

I	1. Prepare a Mark List for Students by using Conditional Formatting. 2.Arrange data in Ascending & descending Order 3.Prepare a Calendar by using Auto Formatting 4.Creating Chart 5.Prepare an inventory bill for a company (use Freeze Panel, track changes). 6.Working with Functions like Absolute, writing conditional Expressions(f), and using Logical functions (AND, OR, NOT) 7.Data Validation (Number, Date & Time Validation) 8.Working with Report using Pivot tables. 9.Macros-Relative & Absolute Macros 10.Implementing EB Bill by using Excel Formula's.	

Learning Resources:

- **Recommended Tex**

  Excel 2019 All-in-One For Dummies – 2018- Greg Harvey

- **Reference Books**

  Microsoft Excel 2019 Pivot Table Data Crunching-2019,Bill Jelen and Michael Alexander

- **Web resources:** Web resources from NDL Library, E-content from open source libraries

Second Year

(Semester – III)

Course Code-CC5	**Python Programming**		**Credits 5**
Lecture Hours: (L) per week - 5	**Tutorial Hours : (T) per week**	**Lab Practice Hours: (P)per week**	**Total: (L+T+P) per week: 5**
Course Category : Core	**Year & Semester: II Year III Semester**	**Admission Year:**	
Pre-requisite	Basic Knowledge of Programming concept		

Learning Objectives: (for teachers: what they have to do in the class/lab/field)

- Describe the core syntax and semantics of Python programming language.
- Discover the need for working with the strings and functions.
- Illustrate the process of structuring the data using lists, dictionaries, tuples and sets.
- Understand the usage of packages and Dictionaries

Course Outcomes: (for students: To know what they are going to learn)

CO1: Develop and execute simple Python programs

CO2: Write simple Python programs using conditionals and looping for solving problems

CO3: Decompose a Python program into functions

CO4: Represent compound data using Python lists, tuples, dictionaries etc.

CO5: Read and write data from/to files in Python programs

Recap: (not for examination) Motivation/previous lecture/ relevant portions required for the course) [This is done during 2 Tutorial hours)

Units	Contents	Required Hours
I	**Basics of Python Programming:** History of Python-Features of Python-Literal-Constants-Variables - Identifiers–Keywords-Built-in Data Types-Output	15

	Statements –Input Statements-Comments –Indentation-Operators-Expressions-Type conversions. **Python Arrays:** Defining and Processing Arrays–Array methods.	
II	**Control Statements:** Selection/Conditional Branching statements: if, if-else, nested if and if-else statements. Iterative Statements: while loop, for loop, else suite in loop and nested loops. **Jump Statements:** break, continue and pass statements**.**	**15**
III	**Functions:** Function Definition – Function Call – Variable Scope and its Life time- Return Statement. **Function Arguments**: Required Arguments, Keyword Arguments, Default Arguments and Variable Length Arguments Recursion. **Python Strings:** String operations- Immutable Strings - Built-in String Methods and Functions - String Comparison. **Modules**: import statement- The Python module – dir.() function – Modules and Namespace–Defining our own modules.	**15**
IV	**Lists**: Creating a list –Access values in List-Updating values in Lists-Nested lists-Basic list operations-List Methods. **Tuples**: Creating, Accessing ,Updating and Deleting Elements in a tuple – Nested tuples –Difference between lists and tuples. **Dictionaries and Sets**: Dictionary type in Python - Set Data type.	**15**
V	**Python File Handling** :Types of files in Python - Opening and Closing files-Reading and Writing files: write( ) and write lines( ) methods- append()method–read( )and readlines( )methods–with keyword– Splitting	**15**

	words –File methods-File Positions-Renaming and deleting files.	
Extended Professional Component (is a part of internal component only, Not to be included in the External Examination question paper)	Questions related to the above topics, from various competitive examinations UPSC / TRB / NET / UGC – CSIR / GATE / TNPSC / others to be solved (To be discussed during the Tutorial hour)	
Skills acquired from the course	Knowledge, Problem Solving, Analytical ability, Professional Competency, Professional Communication and Transferrable Skill	

Learning Resources:

- **Recommended Texts**

  1. Charles Dierbach, "Introduction to Computer Science using Python - A computational Problem solving Focus", Wiley India Edition, 2015.

  2. Wesley J. Chun, "Core Python Applications Programming", 3rd Edition , Pearson Education, 2016

- **Reference Books**

  1. Mark Lutz, "Learning Python Powerful Object Oriented Programming", O'reilly Media 2018, 5th Edition.

2. Timothy A. Budd, "Exploring Python", Tata MCGraw Hill Education Private Limited 2011, 1 st Edition.
3. John Zelle, "Python Programming: An Introduction to Computer Science", Second edition, Course Technology Cengage Learning Publications, 2013, ISBN 978- 1590282410
4. Michel Dawson, "Python Programming for Absolute Beginers" , Third Edition, Course Technology Cengage Learning Publications, 2013, ISBN 978-1435455009

- **Web resources**

https://onlinecourses.swayam2.ac.in/cec22_cs20/preview

Course Code: CC6	**Python Programming Lab**		**Credits : 5**
Lecture Hours: (L) per week:	**Tutorial Hours: (T) per week**	**Lab Practice Hours: 5 per week**	**Total: (L+T+P) per week: 5**
Course Category :Core	**Year & Semester: II Year III Semester**		**Admission Year:**
Pre-requisite	Basic of programming skill		

Learning Objectives: (for teachers: what they have to do in the class/lab/field)

- Acquire programming skills in core Python.
- Acquire Object-oriented programming skills in Python.
- Develop the skill of designing graphical-user interfaces (GUI) in Python.
- Develop the ability to write database applications in Python.
- Acquire Python programming skills to move into specific branches

Course Outcomes: (for students: To know what they are going to learn)

CO1: To understand the problem solving approaches

CO2: To learn the basic programming constructs in Python

CO3: To practice various computing strategies for Python-based solutions to real world problems

CO4: To use Python data structures - lists, tuples, dictionaries. **CO5:** To do input/output with files in Python.		
Recap: (not for examination) Motivation/previous lecture/ relevant portions required for the course) [This is done during 2 Tutorial hours)		
	List of Exercises:	**Required Hours**
	1. Program to convert the given temperature from Fahrenheit to Celsius and vice versa depending upon user's choice. 2. Program to calculate total marks, percentage and grade of a student. Marks obtained in each of the five subjects are to be input by user. Assign grades according to the following criteria: Grade A: Percentage >=80 Grade B: Percentage >=70 and 80 Grade C: Percentage >=60 and <70 Grade D: Percentage >=40 and <60 Grade E: Percentage < 40 3. Program to find the area of rectangle, square, circle and triangle by accepting suitable input parameters from user. 4. Write a Python script that prints prime numbers less than 20. 5. Program to find factorial of the given number using recursive function. 6. Write a Python program to count the number of even and odd numbers from array of N numbers. 7. Write a Python class to reverse a string word by word. 8. Given a tuple and a list as input, write a program to count the occurrences of all items of the list in the	

	tuple. (Input : tuple = ('a', 'a', 'c', 'b', 'd'), list = ['a', 'b'], Output : 3) 9. Write a Python program to construct the following pattern, using a nested loop * ** *** **** ***** **** *** ** * 10. Program using Dictionaries. 11. Program using Set. 12. Read a file content and copy only the contents at odd lines into a new file.	
Extended Professional Component (is a part of internal component only, Not to be included in the External Examination question	Questions related to the above topics, from various competitive examinations UPSC / TRB / NET / UGC – CSIR / GATE / TNPSC / others to be solved (To be discussed during the Tutorial hour)	

paper)		
Skills acquired from the course	Knowledge, Problem Solving, Analytical ability, Professional Competency, Professional Communication and Transferrable Skill	

Learning Resources:

- **Recommended Texts**

1. Charles Dierbach, "Introduction to Computer Science using Python - A computational Problem solving Focus", Wiley India Edition, 2015.

2. Wesley J. Chun, "Core Python Applications Programming", 3rd Edition , Pearson Education, 2016

- **Reference Books**

1. Mark Lutz, "Learning Python Powerful Object Oriented Programming", O'reilly Media 2018, 5th Edition.
2. Timothy A. Budd, "Exploring Python", Tata MCGraw Hill Education Private Limited 2011, 1 st Edition.
3. John Zelle, "Python Programming: An Introduction to Computer Science", Second edition, Course Technology Cengage Learning Publications, 2013, ISBN 978- 1590282410
4. Michel Dawson, "Python Programming for Absolute Beginers" , Third Edition, Course Technology Cengage Learning Publications, 2013, ISBN 978-1435455009

Course Code: EC3	**DISCRETE MATHEMATICAL STRUCTURES**		**Credits: 3**
Lecture Hours: (L) per week: 4	**Tutorial Hours : (T) per week**	**Lab Practice Hours: (P)per week**	**Total: (L+T+P) per week: 4**
Course Category : Elective	**Year & Semester: II Year III Semester**		**Admission Year:**
Pre-requisite	Basic Knowledge on probability and mathematical logic		

Learning Objectives: (for teachers: what they have to do in the class/lab/field)

To understand the mathematical concepts like set theory, logics, number theory, combinatory and relations.

Course Outcomes: (for students: To know what they are going to learn)

CO1: To gain knowledge on set theory

CO2: Able to understand different mathematical logics and functions

CO3: To get an idea on Permutations and Combinations

CO4: Understanding the different form of number theory

CO5: Able to understand Relations and its applications

Recap: (not for examination) Motivation/previous lecture/ relevant portions required for the course) [This is done during 2 Tutorial hours)

Units	**Contents**	**Required Hours**
I	***SET THEORY*** Introduction- set and Its Element – Set Description (Roster, Set Builder and cardinal number method) Types of Sets- Set Operations and Laws of set Theory. Partition of sets. Minsets-Countable and un Countable	

	set. Algebra of sets and Duality	
II	***MATHEMATICAL LOGIC*** Basic Logic and Proof, logical operations – Logic Propositional equivalence, Predicates and Quantities, Tautology-Contradiction-Methods of proofs(Direct and Indirect)- Function- Definition-Notation- Types of Function- Composition of Functions-	
III	***NUMBER THEORY*** The Integers and Division, Integers and Algorithms,(Multiplication, Addition and Division -Sequences and Summations, Recursive algorithms, Program correctness	
IV	***COMBINATORICS*:** The basics of counting, the pigeonhole principle, Permutations and Combinations, Binomial coefficients, Generalized permutations and combinations	
V	***RELATIONS*** Relations – Relations and their properties, Representing Relations, Closures of relations, Equivalence relations, Partial orderings-Recurrence Relations Binary Relations.	
Extended Professional Component (is a part of internal component only, Not to	Questions related to the above topics, from various competitive examinations UPSC / TRB / NET / UGC – CSIR / GATE / TNPSC / others to be solved (To be discussed during the Tutorial hour)	

be included in the External Examination question paper)		
Skills acquired from the course	Knowledge, Problem Solving, Analytical ability, Professional Competency, Professional Communication and Transferrable Skill	

Learning Resources:

- **Recommended Texts**

1. Rosen K.H. Discrete Mathematics and its Applications, 5th edition, Tata McGraw – Hills, 2003.
2. J.K Sharma "DISCRETE MATHEMATICS" 3 rd Edition Macmillan Reprint 2011

- **Reference Books**

1. Johnson Baugh R, and Carman R, Discrete mathematics, 5th edition, Person Education, 2003.
2. Kolman B, Busoy R.C, and Ross S.C, Discrete Mathematical Structures, 5th edition, Pretitice – Hall, 2004.
3. Mott J.L, Kandel A, and Bake T.P, Discrete Mathematics for Computer Scientists & Mathematicians, 2nd edition, Prentice-Hall of India, 2002.

- **Web resources:** Web resources from NDL Library, E-content from open-source libraries

Course Code: SEC-IV	**Web Application Development**		**Credits: 2**
Lecture Hours: (L) per week: 2	**Tutorial Hours : (T) per week**	**Lab Practice Hours: (P)per week**	**Total: (L+T+P) per week: 2**
Course Category :Skill Enhancement	**Year & Semester: II Year III Semester**	**Admission Year:**	
Pre-requisite			

- Insert a graphic with in a webpage.
- Create a link with in a webpage.
- Create a table with in a webpage.
- Insert heading levels within a webpage.
- Insert ordered and unordered lists with in a webpage .Create a webpage.

On completion of this course, students will

CO1 : Knows the basic concept in HTML Concept of resources in HTML

CO2: Knows Design concept.

CO3: Understand the page formatting. Concept of list

CO4 : Creating Links. Know the concept of creating link to email address

CO5 : Concept of adding images and Understand the table creation.

UNIT	Contents	Required Hours
I	Introduction : Web Basics: What is Internet– Web browsers–What is Webpage –HTML Basics:	**6**

	Understanding tags.	
II	Tags for Document structure (HTML, Head , Body Tag). Block level text elements: Headings paragraph(<p>tag)– Font style elements:(bold, italic, font, small, strong, strike, big tags)	**6**
III	Lists: Types of lists: Ordered, Unordered–Nesting Lists – Other tags: Marquee, HR,BR-Using Images–Creating Hyperlinks.	**6**
IV	Tables: Creating basic Table, Table elements, Caption– Table and cell alignment– Row span, Colspan –Cell padding.	**6**
V	Frames: Frameset– Targeted Links–No frame–Forms: Input, Text area, Select, Option.	**6**

Learning Resources:

1. **Mastering HTML5 and CSS3 MadeEasy‖,TeachUCompInc.,2014.**
2. **Thomas Michaud "Foundations of Web Design: Introduction to HTML & CSS"**

Web resources

1. https://www.teachucomp.com/samples/html/5/manuals/Mastering-HTML5-CSS3.pdf
2. https://www.w3schools.com/html/default.asp.

Course Code: SEC-V	**Web Application Development Lab**		**Credits: 2**
Lecture Hours: (L) per week: -	**Tutorial Hours : (T) per week**	**Lab Practice Hours: 2 per week**	**Total: (L+T+P) per week: 2**

Course Category :Skill Enhancement	**Year & Semester: II Year III Semester**	**Admission Year:**
Pre-requisite		

- Insert a graphic with in a webpage.
- Create a link with in a webpage.
- Create a table with in a webpage.
- Insert heading levels within a webpage.
- Insert ordered and unordered lists with in a webpage .Create a webpage.

On completion of this course, students will

CO1 : Knows the basic concept in HTML Concept of resources in HTML

CO2: Knows Design concept.

CO3: Understand the page formatting. Concept of list

CO4 : Creating Links. Know the concept of creating link to email address

CO5 : Concept of adding images and Understand the table creation.

UNIT	**Contents**	**Required Hours**
	1. Write a HTML code to create a web page with any color background and display moving message in any color. 2. Write an HTML code to display your CV on a web page. 3. Write an HTML code to create a Home page for college 4. Write an HTML code to illustrate the usage of the following: • Ordered List • Unordered List • Definition List	

	5. Write an HTML code to create a Home page for college having three links: About Us, Our Department and Contact Us. Create separate web pages for the three links. 6.Write a HTML code to design a Greeting card using image tag. 7. Write a HTML code to display mark sheet using Table tag. 8. Write a HTML code to display using Table and cell alignment attributes. 9. Create a web page which divides the page in two equal frames and place images in frame-1 and frame-2 respectively. 10. Write a HTML program to develop a static Registration Form.	

Learning Resources:

1. **Mastering HTML5 and CSS3 MadeEasy‖,TeachUCompInc.,2014.**

2 .**Thomas Michaud "Foundations of Web Design: Introduction to HTML & CSS"**

Web resources

3. https://www.teachucomp.com/samples/html/5/manuals/Mastering-HTML5-CSS3.pdf
4. https://www.w3schools.com/html/default.asp.

Second Year

(Semester – IV)

Course Code: CC7	**Java Programming**	**Credits: 5**

Lecture Hours: (L) per week: 5	**Tutorial Hours : (T) per week**	**Lab Practice Hours: (P)per week**	**Total: (L+T+P) per week: 5**
Course Category :Core	**Year & Semester: II Year IV Semester**		**Admission Year:**
Pre-requisite	Basic Programming skill		

Learning Objectives: (for teachers: what they have to do in the class/lab/field)

- To provide fundamental knowledge of object-oriented programming.
- To equip the student with programming knowledge in Core Java from the basics up.
- To enable the students to use AWT controls, Event Handling and Swing for GUI.

Course Outcomes: (for students: To know what they are going to learn)

CO1: Understand the basic Object-oriented concepts.

Implement the basic constructs of Core Java

CO2: Implement inheritance, packages, interfaces and exception handling of Core Java.

CO3: Implement multi-threading and I/O Streams of Core Java

CO4: Implement AWT and Event handling.

CO5: Use Swing to create GUI.

Recap: (not for examination) Motivation/previous lecture/ relevant portions required for the course) [This is done during 2 Tutorial hours)

Units	**Contents**	**Required Hours**
I	**Introduction:** Review of Object Oriented concepts - History of Java - Java buzzwords - JVM architecture - Data types - Variables - Scope and life time of variables - arrays - operators - control statements - type conversion and casting - simple java program - constructors - methods - Static block - Static Data - Static Method String and String Buffer Classes	**15**
II	**Inheritance**: Basic concepts - Types of inheritance - Member access rules - Usage of this and Super key word - Method	**15**

	Overloading - Method overriding - Abstract classes - Dynamic method dispatch - Usage of final keyword. **Packages**: Definition - Access Protection - Importing Packages. **Interfaces**: Definition – Implementation – Extending Interfaces. **Exception Handling**: *try – catch - throw - throws – finally* – Built-in exceptions - Creating Own Exception classes.	
III	**Multithreaded Programming**: Thread Class - Runnable interface – Synchronization – Using synchronized methods – Using *synchronized* statement - Interthread Communication – Deadlock. **I/O Streams:** Concepts of streams - Stream classes- Byte and Character stream - Reading console Input and Writing Console output - File Handling.	**15**
IV	**AWT Controls:** The AWT class hierarchy - user interface components- Labels - Button - Text Components - Check Box - Check Box Group - Choice - List Box - Panels – Scroll Pane - Menu - Scroll Bar. Working with Frame class - Colour - Fonts and layout managers. **Event Handling:** Events - Event sources - Event Listeners - Event Delegation Model (EDM) - Handling Mouse and Keyboard Events - Adapter classes - Inner classes.	**15**
V	**Swing:** Introduction to Swing - Hierarchy of swing components. Containers - Top level containers - JFrame - JWindow - JDialog - JPanel - JButton - JToggleButton -	**15**

	JCheckBox - JRadioButton - JLabel,JTextField - JTextArea - JList - JComboBox - JScrollPane	
Extended Professional Component (is a part of internal component only, Not to be included in the External Examination question paper)	Questions related to the above topics, from various competitive examinations UPSC / TRB / NET / UGC – CSIR / GATE / TNPSC / others to be solved (To be discussed during the Tutorial hour)	
Skills acquired from the course	Knowledge, Problem Solving, Analytical ability, Professional Competency, Professional Communication and Transferrable Skill	

Learning Resources:

- **Recommended Texts**
  1. Herbert Schildt, The Complete Reference, Tata McGraw Hill, New Delhi, 7th Edition, 2010.
  2. Gary Cornell, Core Java 2 Volume I – Fundamentals, Addison Wesley, 1999.
- **Reference Books**

1. Head First Java, O'Rielly Publications,
2. Y. Daniel Liang, *Introduction to Java Programming*, 7th Edition, Pearson Education India, 2010.

Web resources: Web resources from NDL Library, E-content from open-source libraries

Course Code: CC8	**Java Programming Lab**		**Credits: 5**
Lecture Hours: (L) per week	**Tutorial Hours : (T) per week**	**Lab Practice Hours: 5 per week**	**Total: (L+T+P) per week: 5**
Course Category :Core	**Year & Semester: II Year IV Semester**		**Admission Year:**
Pre-requisite	Basic Programming debugging skills		

Learning Objectives: (for teachers: what they have to do in the class/lab/field)

- To gain practical expertise in coding Core Java programs
- To become proficient in the use of AWT, Event Handling and Swing.

Course Outcomes: (for students: To know what they are going to learn)

CO1: Code, debug and execute Java programs to solve the given problems

CO2: Implement multi-threading and exception-handling

CO3: Implement functionality using String and StringBuffer classes

CO4: Demonstrate Event Handling.

CO5: Create applications using Swing and AWT

Recap: (not for examination) Motivation/previous lecture/ relevant portions required for the

course) [This is done during 2 Tutorial hours)		
	List of Exercises:	**Required Hours**
	1. Write a Java program that prompts the user for an integer and then prints out all the prime numbers up to that Integer? 2. Write a Java program to multiply two given matrices. 3. Write a Java program that displays the number of characters, lines and words in a text? 4. Generate random numbers between two given limits using Random class and print messages according to the range of the value generated. 5. Write a program to do String Manipulation using Character Array and perform the following string operations: a. String length b. Finding a character at a particular position c. Concatenating two strings 6. Write a program to perform the following string operations using String class: a. String Concatenation b. Search a substring c. To extract substring from given string 7. Write a program to perform string operations using StringBuffer class: a. Length of a string	

	b. Reverse a string c. Delete a substring from the given string 8. Write a java program that implements a multi-thread application that has three threads. First thread generates random integer every 1 second and if the value is even, second thread computes the square of the number and prints. If the value is odd, the third thread will print the value of cube of the number. 9. Write a threading program which uses the same method asynchronously to print the numbers 1 to 10 using Thread1 and to print 90 to 100 using Thread2. 10. Write a program to demonstrate the use of following exceptions. a. Arithmetic Exception b. Number Format Exception c. Array Index Out of Bound Exception d. Negative Array Size Exception 11. Write a Java program that reads on file name from the user, then displays information about whether the file exists, whether the file is readable, whether the file is writable, the type of file and the length of the file in bytes.	

Extended Professional Component (is a part of internal component only, Not to be included in the External Examination question paper)	Questions related to the above topics, from various competitive examinations UPSC / TRB / NET / UGC – CSIR / GATE / TNPSC / others to be solved (To be discussed during the Tutorial hour)	
Skills acquired from the course	Knowledge, Problem Solving, Analytical ability, Professional Competency, Professional Communication and Transferrable Skill	

Learning Resources:

Learning Resources:

- **Recommended Texts**

  1.Herbert Schildt, The Complete Reference, Tata McGraw Hill, New Delhi, 7th Edition, 2010.

  2.Gary Cornell, Core Java 2 Volume I – Fundamentals, Addison Wesley, 1999.

- **Reference Books**

  1.Head First Java, O'Rielly Publications,

  2. Y. Daniel Liang, *Introduction to Java Programming*, 7th Edition, Pearson Education India, 2010.

Web resources: Web resources from NDL Library, E-content from open-source libraries

Course Code- **EC4**	**NUMERICAL METHODS**		Credits 3
Lecture Hours: (L) per week - 5	Tutorial Hours :75 (T) per week	Lab Practice Hours: (P)per week	Total: (L+T+P) per week: 5
Course Category : Elective	Year & Semester: II Year IV Semester	Admission Year:	
Pre-requisite	Basic Knowledge of Programming concept		

The main objectives of this course are:

1. To introduce the various topics in Numerical methods.
2. To make understand the fundamentals of algebraic equations.
3. To apply interpolation and approximation on examples.
4. To solve problems using numerical differentiation and integration.
5. To solve linear systems, numerical solution of ordinary differential equations.

Course Outcomes: (for students: To know what they are going to learn)

CO1: Know how to solve various problems on numerical methods

CO2: Use approximation to solve problems

CO3: Differentiation and integration concept are applied

CO4: Apply , direct methods for solving linear systems

CO5: Numerical solution of ordinary differential equations

Recap: (not for examination) Motivation/previous lecture/ relevant portions required for the course) [This is done during 2 Tutorial hours)

Units	Contents	Required Hours
I	FUNDAMENTALS OF ALGEBRAIC EQUATION: Solution of algebraic and transcendental equations-Bisection method – Fixed point iteration method – Newton Raphson method – Gauss elimination method – Gauss Jordan method .	15
II	ITERATIVE, INTERPOLATION AND APPROXIMATION: Iterative methods - Gauss Jacobi and	15

	Gauss Seidel – Jacobi's method for symmetric matrices. Interpolation with unequal intervals – Lagrange's interpolation – Newton's divided difference interpolation	
III	INTERPOLATION WITH EQUAL INTERVAL: Difference operators and relations. -Interpolation with equal intervals – Newton's forward and backward difference formulae.	15
IV	NUMERICAL DIFFERENTIATION AND INTEGRATION: Approximation of derivatives using interpolation polynomials – Numerical integration using Trapezoidal, Simpson's 1/3 rule and Simpson's 3/8 th rule .	15
V	INITIAL VALUE PROBLEMS FOR ORDINARY DIFFERENTIAL EQUATIONS: Single step methods – Taylor's series method – Euler's method – Modified Euler's method - Runge Kutta method for solving(first, second , Third and 4th) order equations – Multi step methods	15
Extended Professional Component (is a part of internal component only, Not to be included in the External Examination question paper)	Questions related to the above topics, from various competitive examinations UPSC / TRB / NET / UGC – CSIR / GATE / TNPSC / others to be solved (To be discussed during the Tutorial hour)	

Skills acquired from the course	Knowledge, Problem Solving, Analytical ability, Professional Competency, Professional Communication and Transferrable Skill	

Learning Resources:

- Recommended Texts

  1. Charles Dierbach, "Introduction to Computer Science using Python - A computational Problem solving Focus", Wiley India Edition, 2015.

  2. Wesley J. Chun, "Core Python Applications Programming", 3rd Edition , Pearson Education, 2016

- Reference Books

  1. Mark Lutz, "Learning Python Powerful Object Oriented Programming", O'reilly Media 2018, 5th Edition.
  2. Timothy A. Budd, "Exploring Python", Tata MCGraw Hill Education Private Limited 2011, 1 st Edition.
  3. John Zelle, "Python Programming: An Introduction to Computer Science", Second edition, Course Technology Cengage Learning Publications, 2013, ISBN 978- 1590282410
  4. Michel Dawson, "Python Programming for Absolute Beginers" , Third Edition, Course Technology Cengage Learning Publications, 2013, ISBN 978-1435455009

- Web resources

  1. https://onlinecourses.swayam2.ac.in/cec22_cs20/preview

Course Code: SEC-6	**PHP Programming**		**Credits: 2**
Lecture Hours: (L) per week: 2	**Tutorial Hours : (T) per week**	**Lab Practice Hours: (P)per week**	**Total: (L+T+P) per week: 2**
Course Category :Skill Enhancement	**Year & Semester: II Year IV Semester**		**Admission Year:**
Pre-requisite	Basic Knowledge on Web		

Learning Objectives: (for teachers: what they have to do in the class/lab/field)

- To knowledge about Server Side PHP Programming for web application.

Course Outcomes: (for students: To know what they are going to learn)

CO1: Analyze the history and configuration of PHP programming.

CO2: Known about GET and POST request with HTML forms.

CO3: Summarize the date and time, string manipulation and regular expression functions. **CO4:** Identify the HTTP header and HTTP cookies.

CO5: Analyze the system functions and file operations for error handling.

Recap: (not for examination) Motivation/previous lecture/ relevant portions required for the course) [This is done during 2 Tutorial hours)

Units	**Contents**	**Required Hours**
I	Introduction to PHP -Basic Knowledge of websites -Introduction of Dynamic Website -Introduction to PHP -Scope of PHP -XAMPP and WAMP Installation- PHP Programming Basics -Syntax of PHP -Embedding PHP in HTML -Embedding HTML in PHP.	**06**
II	Introduction to PHP Variable -Understanding Data Types -Using Operators -Using Conditional Statements -If (),else if () and else if condition Statement -Switch () Statements -Using the while () Loop -Using the for() Loop	**06**

III	PHP Functions -PHP Functions -Creating an Array - Modifying Array Elements -Processing Arrays with Loops - Grouping Form Selections with Arrays -Using Array Functions -Using Predefined PHP Functions -Creating User-Defined Functions	**06**
IV	PHP Advanced Concepts -Reading and Writing Files -Reading Data from a File.	**06**
V	Managing Sessions and Using Session Variables -Destroying a Session -Storing Data in Cookies -Setting Cookies.	**06**
Extended Professional Component (is a part of internal component only, Not to be included in the External Examination question paper)	Questions related to the above topics, from various competitive examinations UPSC / TRB / NET / UGC – CSIR / GATE / TNPSC / others to be solved (To be discussed during the Tutorial hour)	
Skills acquired from the course	Knowledge, Problem Solving, Analytical ability, Professional Competency, Professional Communication and Transferrable Skill	

Learning Resources:

- **Recommended Texts**

  Head First PHP & MySQL: A Brain-Friendly Guide- 2009-Lynn mighley and Michael Morrison.

- **Reference Books**

  The Joy of PHP: A Beginner's Guide to Programming Interactive Web Applications with PHP and MySQL- Alan Forbes

Web resources: Web resources from NDL Library, E-content from open-source libraries

Course Code: SEC-7	**PHP Programming Lab**		**Credits: 2**
Lecture Hours: (L) per week: -	**Tutorial Hours : (T) per week**	**Lab Practice Hours: 2 per week**	**Total: (L+T+P) per week: 2**
Course Category :Skill Enhancement	**Year & Semester: II Year IV Semester**		**Admission Year:**
Pre-requisite	Basic Knowledge on Web		

Learning Objectives: (for teachers: what they have to do in the class/lab/field)

- The objective of this course is The students will be able to enhance their analyzing and problem solving skills and use the same for writing programs in PHP.

Course Outcomes: (for students: To know what they are going to learn)

CO1: Write PHP code to produce outcomes and solve problems.

CO2: Display and insert data using PHP and MySQL.

CO3: Test, debug, and deploy web pages containing PHP and MySQL.

Recap: (not for examination) Motivation/previous lecture/ relevant portions required for the

course) [This is done during 2 Tutorial hours)		
	List of Exercise	**Required Hours**
	1.Create a PHP program to demonstrate the get and post method. 2. Create a PHP program to validate the user form. 3.Create a PHP program to demonstrate the different predefined function in array. 4.Create a PHP program to demonstrate the different predefined function in Math. 5.Create a PHP program to demonstrate the different predefined function in Date 6. Write a PHP program to store current date-time in a COOKIE and display the “Last visited on‟ date-time on the web page upon reopening of the same page. 7. Write a PHP program to store current date-time in a COOKIE and display the “Last visited on‟ date-time on the web page upon reopening of the same page. 8. Assume four users user1, user2, user3 and user4 having the passwords pwd1, pwd2, pwd3 and pwd4 respectively. Write a PHP for doing the following. i). Create a Cookie and add these four user ID’s and passwords to this Cookie. ii). Read the user id and passwords entered in the Login form (week1) and authenticate with the values (user id and passwords) available in the cookies. If he is a valid user (i.e., user-name and password match) you should welcome him by name (user-name) else you should display “You are not an authenticated user ’’.	

Extended Professional Component (is a part of internal component only, Not to be included in the External Examination question paper)	Questions related to the above topics, from various competitive examinations UPSC / TRB / NET / UGC – CSIR / GATE / TNPSC / others to be solved (To be discussed during the Tutorial hour)	
Skills acquired from the course	Knowledge, Problem Solving, Analytical ability, Professional Competency, Professional Communication and Transferrable Skill	

Learning Resources:

- **Recommended Texts**

  Head First PHP & MySQL: A Brain-Friendly Guide- 2009-Lynn mighley and Michael Morrison.

- **Reference Books**

  The Joy of PHP: A Beginner's Guide to Programming Interactive Web Applications

with PHP and MySQL- Alan Forbes **Web resources:** Web resources from NDL Library, E-content from open-source libraries

Third Year

(Semester – V)

Course Code: CC9	**Operating Systems**		**Credits: 4**
Lecture Hours: (L) per week: 5	**Tutorial Hours : (T) per week**	**Lab Practice Hours: (P)per week**	**Total: (L+T+P) per week: 5**
Course Category : Core	**Year & Semester: III Year V Semester**		**Admission Year:**
Pre-requisite	Basic Knowledge on Computer and its functions		
Learning Objectives: (for teachers: what they have to do in the class/lab/field) ● Understanding the design of the Operating System ● Imparting knowledge on CPU scheduling, Process and Memory Management. ● To code specialized programs for managing overall resources and operations of the computer.			

Course Outcomes: (for students: To know what they are going to learn)

CO1: Define the fundamentals of OS and identify the concepts relevant to process, process life cycle, Scheduling Algorithms, Deadlock and Memory management

CO2: know the critical analysis of process involving various algorithms, an exposure to threads and semaphores

CO3: Have a complete study about Deadlock and its impact over OS. Knowledge of handling Deadlock with respective algorithms and measures to retrieve from deadlock.

CO4: Have complete knowledge of Scheduling Algorithms and its types.

CO5: understand memory organization and management

Recap: (not for examination) Motivation/previous lecture/ relevant portions required for the course) [This is done during 2 Tutorial hours)

Units	Contents	Required Hours
I	**Introduction**: operating system, history (1990s to 2000 and beyond), distributed computing, parallel computation. **Process concepts:** definition of process, process states-Life cycle of a process, process management- process state transitions, process control block(PCB), process operations, suspend and resume, context switching, Interrupts -Interrupt processing, interrupt classes, Inter process communication-signals, message passing.	**12**
II	**Asynchronous concurrent processes:** mutual exclusion-critical section, mutual exclusion primitives, implementing mutual exclusion primitives, Peterson's algorithm, software solutions to the mutual Exclusion Problem-, n-thread mutual exclusion- Lamports Bakery Algorithm. Semaphores – Mutual exclusion with Semaphores, thread synchronization with semaphores, counting semaphores, implementing semaphores.	**12**

	Concurrent programming: monitors, message passing	
III	**Deadlock and indefinite postponement:** Resource concepts, four necessary conditions for deadlock, deadlock prevention, deadlock avoidance and Dijkstra's Banker's algorithm, deadlock detection, deadlock recovery	**12**
IV	**Job and processor scheduling:** scheduling levels, scheduling objectives, scheduling criteria, preemptive vs non-preemptive scheduling, interval timer or interrupting clock, priorities, scheduling algorithms- FIFO scheduling, RR scheduling, quantum size, SJF scheduling, SRT scheduling, HRN scheduling, multilevel feedback queues, Fair share scheduling	**12**
V	**Real Memory organization and Management**: Memory organization, Memory management, Memory hierarchy, Memory management strategies, contiguous vs non-contiguous memory allocation, single user contiguous memory allocation, fixed partition multiprogramming, variable partition multiprogramming, Memory swapping **Virtual Memory organization:** virtual memory basic concepts, multilevel storage organization, block mapping, paging basic concepts, segmentation, paging/segmentation systems. **Virtual Memory Management:** Demand Paging, Page replacement strategies	**12**

Extended Professional Component (is a part of internal component only, Not to be included in the External Examination question paper)	Questions related to the above topics, from various competitive examinations UPSC / TRB / NET / UGC – CSIR / GATE / TNPSC / others to be solved (To be discussed during the Tutorial hour)	
Skills acquired from the course	Knowledge, Problem Solving, Analytical ability, Professional Competency, Professional Communication and Transferrable Skill	

Learning Resources:

- **Recommended Texts**

1. H.M. Deitel, Operating Systems, Third Edition, Pearson Education Asia, 2011

- **Reference Books**

1. William Stallings, Operating System: Internals and Design Principles, Seventh Edition, Prentice-Hall of India, 2012.
2. A. Silberschatz, and P.B. Galvin., Operating Systems Concepts, Nineth Edition, John Wiley &Sons(ASIA) Pte Ltd.,2012

Web resources: Web resources from NDL Library, E-content from open-source libraries

<table>
<tr><td>Course Code: CC-10</td><td colspan="2">Database Management Systems</td><td>Credits:4</td></tr>
<tr><td>Lecture Hours: (L)
per week: 5</td><td>Tutorial Hours :
(T) per week</td><td>Lab Practice
Hours: (P)per week</td><td>Total: (L+T+P)
per week: 5</td></tr>
<tr><td>Course Category :Core</td><td colspan="2">Year & Semester: III YEAR V SEMESTER</td><td>Admission Year:</td></tr>
<tr><td>Pre-requisite</td><td colspan="3">Basic knowledge on Data and its relations</td></tr>
<tr><td colspan="4">Learning Objectives: (for teachers: what they have to do in the class/lab/field)
● To enable the students to learn the designing of data base systems, foundation on the relational model of data and normal forms.
● To understood the concepts of data base management system, design simple Database models
● To learn and understand to write queries using SQL, PL/SQL.</td></tr>
<tr><td colspan="4">Course Outcomes: (for students: To know what they are going to learn)
CO1: Understand the various basic concepts of Data Base System. Difference between file system and DBMS and compare various data models.
CO2: Define the integrity constraints. Understand the basic concepts of Relational Data Model, Entity-Relationship Model.
CO3: Design database schema considering normalization and relationships within database. Understand and construct database using Structured Query Language. Attain a good practical skill of managing and retrieving of data using Data Manipulation Language (DML).
CO4: Classify the different functions and various join operations and enhance the knowledge of handling multiple tables.
CO5: Learn to design Data base operations and implement using PL/SQL programs. Learn basics of PL/SQL and develop programs using Cursors, Exceptions</td></tr>
<tr><td colspan="4">Recap: (not for examination) Motivation/previous lecture/ relevant portions required for the course) [This is done during 2 Tutorial hours)</td></tr>
<tr><td>Units</td><td colspan="2">Contents</td><td>Required Hours</td></tr>
</table>

I	**Database Concepts:** Database Systems - Data Vs Information - Introducing the database -File system - Problems with file system – Database systems. Data models - Importance - Basic Building Blocks - Business rules - Evolution of Data models - Degrees of Data Abstraction	**12**
II	**Design Concepts:** Relational database model - logical view of data-keys -Integrity rules - relational set operators - data dictionary and the system catalog - relationships -data redundancy revisited -indexes - codd's rules. Entity relationship model - ER diagram	**12**
III	**Normalization of Database Tables**: Database tables and Normalization – The Need for Normalization –The Normalization Process – Higher level Normal Form. **Introduction to SQL**: Data Definition Commands – Data Manipulation Commands – SELECT Queries – Additional Data Definition Commands – Additional SELECT Query Keywords – Joining Database Tables.	**12**
IV	**Advanced SQL**: Relational SET Operators: UNION – UNION ALL – INTERSECT - MINUS.SQL Join Operators: Cross Join – Natural Join – Join USING Clause – JOIN ON Clause – Outer Join. **Sub Queries and Correlated Queries**: WHERE – IN – HAVING – ANY and ALL – FROM. SQL Functions: Date and Time Function – Numeric Function – String Function – Conversion Function	**12**
V	**PL/SQL**: A Programming Language: History – Fundamentals – Block Structure – Comments – Data Types – Other Data Types – Variable Declaration – Assignment operation – Arithmetic operators. **Control Structures and Embedded**	**12**

	SQL: Control Structures – Nested Blocks – SQL in PL/SQL – Data Manipulation – Transaction Control statements. **PL/SQL Cursors and Exceptions**: Cursors – Implicit Cursors, Explicit Cursors and Attributes – Cursor FOR loops – SELECT…FOR UPDATE – WHERE CURRENT OF clause – Cursor with Parameters – Cursor Variables – Exceptions – Types of Exceptions.	
Extended Professional Component (is a part of internal component only, Not to be included in the External Examination question paper)	Questions related to the above topics, from various competitive examinations UPSC / TRB / NET / UGC – CSIR / GATE / TNPSC / others to be solved (To be discussed during the Tutorial hour)	
Skills acquired from the course	Knowledge, Problem Solving, Analytical ability, Professional Competency, Professional Communication and Transferrable Skill	

Learning Resources:

- **Recommended Texts**
  1. Coronel, Morris, Rob, "Database Systems, Design, Implementation and Management", Ninth Edition
  2. Nilesh Shah, "Database Systems Using Oracle", 2nd edition, Pearson Education India, 2016

- **Reference Books**

1. Abraham Silberschatz, Henry F.Korth and S.Sudarshan,"Database System Concepts", McGraw Hill International Publication ,VI Edition.
2. Shio Kumar Singh, "Database Systems ", Pearson publications, II Edition

Web resources: Web resources from NDL Library, E-content from open-source libraries

Course Code: CC11	**Operating Systems Lab**		**Credits: 4**
Lecture Hours: (L) per week: --	**Tutorial Hours : (T) per week**	**Lab Practice Hours: 5 per week**	**Total: (L+T+P) per week: 5**
Course Category : Core	**Year & Semester:III Year V Semester**		**Admission Year:**
Pre-requisite	Basic Knowledge on Computer and its functions		

Learning Objectives: (for teachers: what they have to do in the class/lab/field)

- Understanding the UNIX ENVIRONMENT
- Imparting knowledge on SHELL script.
- To code specialized programs for managing overall resources and operations of the computer.

Course Outcomes: (for students: To know what they are going to learn)

CO1: To use Unix operating system for study of operating system concepts.

CO2: To write the code to implement and modify various concepts in operating systems using Unix

	List of Excerise	Required Hours
	1Write a program to count the number of characters in a given string. 2) Write a program to find whether the given year is leap year or not? 3) Write a program to check whether a given number is even or odd. 4) Write a program to find factorial of a given number. 5) Write a program to print all prime numbers between m and n(m<n). 6) Write a shell Script to assign a file permission to the given file using 7) Program for Pattern matching using grep command 8) To compresses a file using gzip and pack commands. 9) Write a menu driven shell script to implement the following Unix/linux commands. a.ps b.pwd c.date d.who e.who am i 10) To find a given pattern in a list of files of current directory using grep and fgrep commands.	

	11) Write a shell script to create two directories and store five files in one directory using the related commands and to transfer all the files to another directory. 12) Write a shell script to accept a file name as input and display whether it exits or not. If it exists, then give the details of its attributes like access permission, its size etc.	

Course Code: CC-12	**DATABASE MANAGEMENT SYSTEMS LAB**		**Credits:4**
Lecture Hours: (L) per week	**Tutorial Hours : (T) per week**	**Lab Practice Hours: (P)per week: 5**	**Total: (L+T+P) per week:5**
Course Category :Core	**Year & Semester: III Year V semester**		**Admission Year:**
Pre-requisite	Basic Knowledge on Database Tools		
Learning Objectives: (for teachers: what they have to do in the class/lab/field) Students can learn various SQL and PL/SQL commands, cursor and various application programs.			

Course Outcomes: (for students: To know what they are going to learn)

CO1: Understand the various basic concepts of Data Base System. Difference between file system and DBMS and compare various data models.

CO2: Define the integrity constraints. Understand the basic concepts of Relational Data Model, Entity-Relationship Model.

CO3: Design database schema considering normalization and relationships within database. Understand and construct database using Structured Query Language. Attain a good practical skill of managing and retrieving of data using Data Manipulation Language (DML).

CO4: Classify the different functions and various join operations and enhance the knowledge of handling multiple tables.

CO5: Learn to design Data base operations and implement using PL/SQL programs. Learn basics of PL/SQL and develop programs using Cursors, Exceptions

Recap: (not for examination) Motivation/previous lecture/ relevant portions required for the course) [This is done during 2 Tutorial hours)

	List of Exercises:	**Required Hours**
	I. SQL 1. DDL COMMANDS 2. DML COMMANDS 3. TCL COMMANDS ***II. PL/SQL*** 4. FIBONACCI SERIES 5. FACTORIAL 6. STRING REVERSE 7. SUM OF SERIES 8. TRIGGER	

	III. CURSOR 9. STUDENT MARK ANALYSIS USING CURSOR ***IV. APPLICATION*** 10. LIBRARY MANAGEMENT SYSTEM 11. STUDENT MARK ANALYSIS	
Extended Professional Component (is a part of internal component only, Not to be included in the External Examination question paper)	Questions related to the above topics, from various competitive examinations UPSC / TRB / NET / UGC – CSIR / GATE / TNPSC / others to be solved (To be discussed during the Tutorial hour)	
Skills acquired from the Course	Knowledge, Problem Solving, Analytical ability, Professional Competency, Professional Communication and Transferrable Skill	

Learning Resources:

- **Recommended Texts**

  1. Coronel, Morris, Rob, "Database Systems, Design, Implementation and Management", Ninth Edition

  2. Nilesh Shah, "Database Systems Using Oracle", 2nd edition, Pearson Education India, 2016

- **Reference Books**

  1. Abraham Silberschatz, Henry F.Korth and S.Sudarshan,"Database System Concepts", McGraw Hill International Publication ,VI Edition.

  2.Shio Kumar Singh, "Database Systems ", Pearson publications, II Edition

  3. Albert Lulushi, "Developing ORACLE FORMS Applications", Prentice Hall ,1997

Web resources: Web resources from NDL Library, E-content from open-source libraries

Course Code: DSE –I	**Computer Networks**		**Credits: 3**
Lecture Hours: (L) per week: 5	**Tutorial Hours : (T) per week**	**Lab Practice Hours: (P)per week**	**Total: (L+T+P) per week: 5**
Course Category : Discipline Specific Elective	**Year & Semester: III Year V Semester**		**Admission Year:**
Pre-requisite	Basic Knowledge on Networking		

Learning Objectives: (for teachers: what they have to do in the class/lab/field)

- To understand the concept of Data communication and Computer network
- To get a knowledge on routing algorithms.
- To impart knowledge about networking and inter networking devices

To gain the knowledge on Security over Network communication

Course Outcomes: (for students: To know what they are going to learn)

CO1: To Understand the basics of Computer Network architecture, OSI and TCP/IP reference models

CO2: To gain knowledge on Telephone systems and Satellite communications

CO3: To impart the concept of Elementary data link protocols

CO4: To analyze the characteristics of Routing and Congestion control algorithms

CO5: To understand network security and define various protocols such as FTP, HTTP, Telnet, DNS

Recap: (not for examination) Motivation/previous lecture/ relevant portions required for the course) [This is done during 2 Tutorial hours)

Units	Contents	Required Hours
I	Introduction – Network Hardware – Software – Reference Models – OSI and TCP/IP Models – Example Networks: Internet, ATM, Ethernet and Wireless LANs - Physical Layer – Theoretical Basis for Data Communication - Guided Transmission Media	**09**
II	Wireless Transmission - Communication Satellites – Telephone System: Structure, Local Loop, Trunks and Multiplexing and Switching. Data Link Layer: Design Issues – Error Detection and Correction.	**09**
III	Elementary Data Link Protocols - Sliding Window Protocols – Data Link Layer in the Internet - Medium Access Layer – Channel Allocation Problem – Multiple Access Protocols – Bluetooth	**09**
IV	Network Layer - Design Issues - Routing Algorithms - Congestion Control Algorithms – IP Protocol – IP Addresses – Internet Control Protocols.	**09**
V	Transport Layer - Services - Connection Management - Addressing, Establishing and Releasing a Connection – Simple Transport Protocol – Internet Transport Protocols	**09**

	(ITP) - Network Security: Cryptography.	
Extended Professional Component (is a part of internal component only, Not to be included in the External Examination question paper)	Questions related to the above topics, from various competitive examinations UPSC / TRB / NET / UGC – CSIR / GATE / TNPSC / others to be solved (To be discussed during the Tutorial hour)	
Skills acquired from the Course	Knowledge, Problem Solving, Analytical ability, Professional Competency, Professional Communication and Transferrable Skill	

Learning Resources:

- **Recommended Texts**
  1. A. S. Tanenbaum, "Computer Networks", 4th Edition, Prentice-Hall of India, 2008.
- **Reference Books**
  1. B. A. Forouzan, "Data Communications and Networking", Tata McGraw Hill, 4th Edition, 2017.
  2. F. Halsall, "Data Communications, Computer Networks and Open Systems", Pearson Education, 2008.
  3. D. Bertsekas and R. Gallagher, "Data Networks", 2nd Edition, PHI, 2008.
  4. Lamarca, "Communication Networks", Tata McGraw- Hill, 2002

Web resources: Web resources from NDL Library, E-content from open-source libraries

Course Code: DSE-II	**Introduction to Data Science**		**Credits: 3**
Lecture Hours: (L) per week:4	**Tutorial Hours : (T) per week**	**Lab Practice Hours: (P)per week**	**Total: (L+T+P) per week:4**
Course Category : Discipline Specific Elective	**Year & Semester: III Year V Semester**		**Admission Year:**
Pre-requisite	Basic knowledge on Data and statistics		

Learning Objectives: (for teachers: what they have to do in the class/lab/field)

- To introduce the concepts, techniques and tools in Data Science
- To understand the various facets of data science practice, including data collection and integration, exploratory data analysis, predictive modeling, descriptive modeling and effective communication.

Course Outcomes: (for students: To know what they are going to learn)

CO1: To describe what Data Science is, what Statistical Inference means, identify probability distributions, fit a model to data and use tools for basic analysis and communication

CO2: To describe what Data Science is, what Statistical Inference means, identify probability distributions, fit a model to data and use tools for basic analysis and communication

CO3: To describe what Data Science is, what Statistical Inference means, identify probability distributions, fit a model to data and use tools for basic analysis and communication

CO4: To describe what Data Science is, what Statistical Inference means, identify probability distributions, fit a model to data and use tools for basic analysis and communication

CO5: To describe what Data Science is, what Statistical Inference means, identify probability distributions, fit a model to data and use tools for basic analysis and communication		
Recap: (not for examination) Motivation/previous lecture/ relevant portions required for the course) [This is done during 2 Tutorial hours)		
Units	**Contents**	**Required Hours**
I	**Introduction:** Benefits and uses – Facets of data – Data science process – Big data ecosystem and data science	**09**
II	**The Data science process**: ● Overview – research goals - retrieving data - transformation – Exploratory Data Analysis – Model building	**09**
III	**Algorithms** : ● Machine learning algorithms – Modeling process – Types – Supervised – Unsupervised - Semi-supervised	**09**
IV	**Introduction to Hadoop** : ● Hadoop framework – Spark – replacing MapReduce– NoSQL – ACID – CAP – BASE – types	**09**
V	**Case Study**:	**09**

	● Prediction of Disease - Setting research goals - Data retrieval – preparation - exploration - Disease profiling - presentation and automation	
Extended Professional Component (is a part of internal component only, Not to be included in the External Examination question paper)	Questions related to the above topics, from various competitive examinations UPSC / TRB / NET / UGC – CSIR / GATE / TNPSC / others to be solved (To be discussed during the Tutorial hour)	
Skills acquired from the Course	Knowledge, Problem Solving, Analytical ability, Professional Competency, Professional Communication and Transferrable Skill	

Learning Resources:

- **Recommended Texts**
    1. Davy Cielen, Arno D. B. Meysman, Mohamed Ali, "Introducing Data Science", manning publications 2016

- **Reference Books**

    1. Roger Peng, "The Art of Data Science", lulu.com 2016.

2. MurtazaHaider, “Getting Started with Data Science – Making Sense of Data with Analytics”, IBM press, E-book.
3. Davy Cielen, Arno D.B. Meysman, Mohamed Ali,“Introducing Data Science: Big Data, Machine Learning, and More, Using Python Tools”, Dreamtech Press 2016.
4. Annalyn Ng, Kenneth Soo, “Numsense! Data Science for the Layman: No Math Added”, 2017,1st Edition.
5. Cathy O'Neil, Rachel Schutt, “Doing Data Science Straight Talk from the Frontline”, O'Reilly Media 2013.
6. Lillian Pierson, “Data Science for Dummies”, 2017 II Edition

Web resources: Web resources from NDL Library, E-content from open-source libraries

Third Year (Semester – VI)

Course Code: CC13	**.Net Programming**		**Credits: 4**
Lecture Hours: (L) per week: 5	**Tutorial Hours : (T) per week**	**Lab Practice Hours: (P)per week**	**Total: (L+T+P) per week:5**
Course Category :Core	**Year & Semester: III Year VI Semester**	**Admission Year:**	
Pre-requisite	Basic knowledge on web programming		
Learning Objectives: (for teachers: what they have to do in the class/lab/field)			

1. To develop ASP.NET Web application using standard controls.
2. To create rich database applications using ADO.NET.
3. To implement file handling operations.
4. To utilize ASP.NET security features for authenticating the web site.
5. To handles SQL Server Database using ADO.NET.

Course Outcomes: (for students: To know what they are going to learn)

CO1: To identify and **understand** the goals and objectives of the .NET framework and ASP.NET with C# language.

CO2: To **develop** web application using various controls.

CO3: To analyze C# programming techniques in developing web applications.

CO4: To assess a Web application using Microsoft ADO.NET.

CO5: To **develop** a software to solve real-world problems using ASP.NET

Recap: (not for examination) Motivation/previous lecture/ relevant portions required for the course) [This is done during 2 Tutorial hours)

Units	Contents	Required Hours
I	Overview of .NET framework: Common Language Runtime (CLR), Framework Class Library- C# Fundamentals: Primitive types and Variables – Operators - Conditional statements -Looping statements – Creating and using Objects – Arrays – String operations.	**12**
II	Introduction to ASP.NET - IDE-Languages supported Components -Working with Web Forms – Web form standard controls: Properties and its events – HTML controls -List Controls: Properties and its events.	**12**
III	Rich Controls: Properties and its events – validation controls: Properties and its events– File Stream classes - File Modes – File Share – Reading and Writing to files – Creating, Moving, Copying and Deleting files – File uploading.	**12**

IV	ADO.NET Overview – Database Connections – Commands – Data Reader - Data Adapter - Data Sets - Data Controls and its Properties - Data Binding	**12**
V	Grid View control: Deleting, editing, Sorting and Paging. XML classes – Web form to manipulate XML files - Website Security - Authentication - Authorization – Creating a Web application.	**12**
Extended Professional Component (is a part of internal component only, Not to be included in the External Examination question paper)	Questions related to the above topics, from various competitive examinations UPSC / TRB / NET / UGC – CSIR / GATE / TNPSC / others to be solved (To be discussed during the Tutorial hour)	
Skills acquired from the course	Knowledge, Problem Solving, Analytical ability, Professional Competency, Professional Communication and Transferrable Skill	

Learning Resources:

- **Recommended Texts**

1. SvetlinNakov,VeselinKolev& Co, Fundamentals of Computer Programming with

C#,Faber publication, 2019.

2. Mathew, Mac Donald, The Complete Reference ASP.NET, Tata McGraw-Hill ,2015.

- **Reference Books**

1. Herbert Schildt, The Complete Reference C#.NET, Tata McGraw-Hill,2017.
2. Kogent Learning Solutions, C# 2012 Programming Covers .NET 4.5 Black Book, Dreamtech pres,2013.
3. Anne Boehm, Joel Murach, Murach's C# 2015, Mike Murach& Associates Inc. 2016.
4. DenielleOtey, Michael Otey, ADO.NET: The Complete reference, McGraw Hill,2008.
5. Matthew MacDonald, Beginning ASP.NET 4 in C# 2010, APRESS,2010.

Web resources: Web resources from NDL Library, E-content from open-source libraries

Course Code: CC14	**.Net Programming Lab**		**Credits: 4**
Lecture Hours: (L) per week	**Tutorial Hours : (T) per week**	**Lab Practice Hours: 5 per week**	**Total: (L+T+P) per week: 5**
Course Category :Core	**Year & Semester: III Year VI Semester**		**Admission Year:**
Pre-requisite	Basic knowledge on		

Learning Objectives: (for teachers: what they have to do in the class/lab/field)

1. To develop ASP.NET Web application using standard controls.

2. To create rich database applications using ADO.NET.
3. To implement file handling operations.
4. To utilize ASP.NET security features for authenticating the web site.
5. To handles SQL Server Database using ADO.NET.

Course Outcomes: (for students: To know what they are going to learn)

CO1: To identify and **understand** the goals and objectives of the .NET framework and ASP.NET with C# language.

CO2: To **develop** web application using various controls.

CO3: To analyze C# programming techniques in developing web applications.

CO4: To assess a Web application using Microsoft ADO.NET.

CO5: To **develop** a software to solve real-world problems using ASP.NET

Recap: (not for examination) Motivation/previous lecture/ relevant portions required for the course) [This is done during 2 Tutorial hours)

	List of Exercises:	**Required Hours**
	1. Create an exposure of Web applications and tools 2. Implement the Html Controls 3. Implement the Server Controls 4. Web application using Web controls. 5. Web application using List controls. 6. Web Page design using Rich control. Validate user input using Validation controls. Working with File concepts. 7. Web application using Data Controls. 8. Data binding with Web controls 9. Data binding with Data Controls. 10. Database application to perform insert, update and delete operations.	

	11. Database application using Data Controls to perform insert, delete, edit, paging and sorting operation. 12. Implement the Xml classes. 13. Implement Authentication – Authorization. 14. Ticket reservation using ASP.NET controls. Online examination using ASP.NET controls	
Extended Professional Component (is a part of internal component only, Not to be included in the External Examination question paper)	Questions related to the above topics, from various competitive examinations UPSC / TRB / NET / UGC – CSIR / GATE / TNPSC / others to be solved (To be discussed during the Tutorial hour)	
Skills acquired from the course	Knowledge, Problem Solving, Analytical ability, Professional Competency, Professional Communication and Transferrable Skill	

Learning Resources:

- **Recommended Texts**

1. SvetlinNakov,VeselinKolev& Co, Fundamentals of Computer Programming with C#,Faber publication, 2019.
2. Mathew, Mac Donald, The Complete Reference ASP.NET, Tata McGraw-Hill ,2015.

- **Reference Books**

1. Herbert Schildt, The Complete Reference C#.NET, Tata McGraw-Hill,2017.

2. Kogent Learning Solutions, C# 2012 Programming Covers .NET 4.5 Black Book, Dreamtech pres,2013.

3. Anne Boehm, Joel Murach, Murach's C# 2015, Mike Murach& Associates Inc. 2016.

4.DenielleOtey, Michael Otey, ADO.NET: The Complete reference, McGraw Hill,2008.

5.Matthew MacDonald, Beginning ASP.NET 4 in C# 2010, APRESS,2010.

Web resources: Web resources from NDL Library, E-content from open-source libraries

Course Code: DSE – III	**Cyber Security**		**Credits: 3**
Lecture Hours: (L) per week 5	**Tutorial Hours: (T) per week**	**Lab Practice Hours: (P)per week**	**Total: (L+T+P) per week: 5**
Course Category : Discipline Specific Elective	**Year & Semester: III Year VI Semester**		**Admission Year:**
Pre-requisite	Basic skills on internet and its functions		

Learning Objectives: (for teachers: what they have to do in the class/lab/field)

The students will be able to

- Understand various block cipher and stream cipher models
- Describe the principles of public key cryptosystems, hash functions and digital signature

- To get a firm knowledge on Cyber Security Essentials

Course Outcomes: (for students: To know what they are going to learn)

CO1: Implement basic security algorithms required by any computing system

CO2: Analyze the vulnerabilities in any computing system and hence be able to design a security solution

CO3: Analyze the possible security attacks in complex real time systems and their effective countermeasures

CO4: Differentiate various governing bodies of cyber laws

CO5: Impart various privacy policies for an organization

Recap: (not for examination) Motivation/previous lecture/ relevant portions required for the course) [This is done during 2 Tutorial hours)

Units	Contents	Required Hours
I	**Introduction to Security** Data Encryption Standard-Block cipher principles-block cipher modes of operation-Advanced Encryption Standard (AES)-Triple DES-Blowfish-RC5 algorithm.	**9**
II	**Public Key Cryptography and Hash Algorithms** Principles of public key cryptosystems-The RSA algorithm-Key management - Diffie Hellman Key exchange- Hash functions-Hash Algorithms (MD5, Secure Hash Algorithm	**9**
III	**Fundamentals of Cyber Security** How Hackers Cover Their Tracks- Fraud Techniques- Threat Infrastructure- Techniques to Gain a Foothold (Shellcode, SQL Injection, Malicious PDF Files)- Misdirection, Reconnaissance, and Disruption Methods.	**9**

IV	**Planning for Cyber Security** Privacy Concepts -Privacy Principles and Policies - Authentication and Privacy - Data Mining - Privacy on the Web - Email Security - Privacy Impacts of Emerging Technologies.	**9**
V	**Cyber Security Management** Security Planning - Business Continuity Planning - Handling Incidents - Risk Analysis - Dealing with Disaster – Legal Issues – Protecting programs and Data – Information and the law – Rights of Employees and Employers - Emerging Technologies - The Internet of Things - Cyber Warfare.	**9**
Extended Professional Component (is a part of internal component only, Not to be included in the External Examination question paper)	Questions related to the above topics, from various competitive examinations UPSC / TRB / NET / UGC – CSIR / GATE / TNPSC / others to be solved (To be discussed during the Tutorial hour)	

Skills acquired from the course	Knowledge, Problem Solving, Analytical ability, Professional Competency, Professional Communication and Transferrable Skill	

Learning Resources:

- **Recommended Texts**

. 1. William Stallings, "Cryptography and Network Security", Pearson Education, 6th Edition,2013.

2. Charles P. Pfleeger Shari Lawrence Pfleeger Jonathan Margulies, Security in Computing, 5th Edition , Pearson Education , 2015.

- **Reference Books**

1. Graham, J. Howard, R., Olson, R., Cyber Security Essentials, CRC Press, 2011.

2. George K.Kostopoulous, Cyber Space and Cyber Security, CRC Press, 2013.

Web resources: Web resources from NDL Library, E-content from open-source libraries

Course Code: DSE-IV	**Software Engineering**		**Credits: 3**
Lecture Hours: (L) per week: 5	**Tutorial Hours : (T) per week**	**Lab Practice Hours: (P)per week**	**Total: (L+T+P) per week: 5**
Course Category : Discipline Specific Elective	**Year & Semester: III Year VI Semester**	**Admission Year:**	
Pre-requisite	Basic Knowledge on Software Applications		

Learning Objectives: (for teachers: what they have to do in the class/lab/field)

• To understand the software engineering concepts and to create a system model in real life applications

Course Outcomes: (for students: To know what they are going to learn)

CO1: Gain basic knowledge of analysis and design of systems

CO2: Ability to apply software engineering principles and techniques

CO3: Model a reliable and cost-effective software system

CO4: Ability to design an effective model of the system

CO5: Perform Testing at various levels and produce an efficient system.

Recap: (not for examination) Motivation/previous lecture/ relevant portions required for the course) [This is done during 2 Tutorial hours)

Units	Contents	Required Hours
I	Introduction: The software engineering discipline, programs vs. software products, why study software engineering, emergence of software engineering, Notable changes in software development practices, computer systems engineering. Software Life Cycle Models: Why use a life cycle model, Classical waterfall model, iterative waterfall model, prototyping model, evolutionary model, spiral model, comparison of different life cycle models.	**09**
II	**Requirements Analysis and Specification:** Requirements gathering and analysis, Software requirements specification (SRS) **Software Design**: Good software design, cohesion and coupling, neat arrangement, software design approaches, object- oriented vs function-oriented design	**09**

III	Function-Oriented Software Design: Overview of SA/SD methodology, structured analysis, data flow diagrams (DFD's), structured design, detailed design. User-Interface design: Characteristics of a good interface; basic concepts; types of user interfaces; component based GUI development, a user interface methodology.	**09**
IV	Coding and Testing: Coding; code review; testing; testing in the large vs testing in the small; unit testing; black-box testing; white-box testing; debugging; program analysis tools; integration testing; system testing; some general issues associated with testing. Software Reliability and Quality Management: Software reliability; statistical testing; software quality; software quality management system; SEI capability maturity model; personal software process.	**09**
V	Computer Aided Software Engineering: CASE and its scope; CASE environment; CASE support in software life cycle; other characteristics of CASE tools; towards second generation CASE tool; architecture of a CASE environment. Software Maintenance: Characteristic of software maintenance; software reverse engineering; software maintenance process models; estimation of maintenance cost;	**09**
Extended Professional Component (is a part of internal component	Questions related to the above topics, from various competitive examinations UPSC / TRB / NET / UGC – CSIR / GATE / TNPSC / others to be solved (To be discussed during the Tutorial hour)	

only, Not to be included in the External Examination question paper)		
Skills acquired from the course	Knowledge, Problem Solving, Analytical ability, Professional Competency, Professional Communication and Transferrable Skill	

Learning Resources:

- **Recommended Texts**
  1. Rajib Mall, Fundamentals of Software Engineering, Fifth Edition, Prentice-Hall of India, 2018
- **Reference Books**
  1. Richard Fairley, Software Engineering Concepts, Tata McGraw-Hill publishing company Ltd, Edition 1997.
  2. Roger S. Pressman, Software Engineering, Seventh Edition, McGraw-Hill.

  James A. Senn, Analysis & Design of Information Systems, Second Edition, McGraw-Hill International Editions.

Web resources: Web resources from NDL Library, E-content from open-source libraries

Course Code: PCSEC	**Cloud Computing**		**Credits: 2**
Lecture Hours: (L) per week 5	**Tutorial Hours : (T) per week**	**Lab Practice Hours: (P)per week**	**Total: (L+T+P) per week: 5**

Course Category : Professional competency Skill enhancement	**Year & Semester: III Year VI Semester**	**Admission Year:**
Pre-requisite	Basic knowledge on virtual storage or cloud concept	

Learning Objectives: (for teachers: what they have to do in the class/lab/field)

- To impart fundamental concepts of Cloud Computing.
- To impart a working knowledge of the various cloud service types and their uses and pitfalls.
- To enable the students to know the common features and differences in the service offerings of the three major Cloud Computing service providers, namely Amazon, Microsoft and Google.
- To provide know-how of the various aspects of application design, benchmarking and security on the Cloud.

Course Outcomes: (for students: To know what they are going to learn)

CO1: To understand the concepts and technologies involved in Cloud Computing.

CO2: To understand the concepts of various cloud services and their implementation in the Amazon, Microsoft and Google cloud computing platforms.

CO3: To understand the aspects of application design for the Cloud.

CO4: To understand the concepts involved in benchmarking and security on the Cloud.

CO5: To understand the way in which the cloud is used in various domains.

Recap: (not for examination) Motivation/previous lecture/ relevant portions required for the course) [This is done during 2 Tutorial hours)

Units	**Contents**	**Required Hours**
I	**Introduction to Cloud Computing:** Definition of Cloud Computing – Characteristics of Cloud Computing – Cloud Models – Cloud Service Examples – Cloud-based Services and Applications. **Cloud Concepts and Technologies:** Virtualization – Load balancing – Scalability and Elasticity – Deployment – Replication – Monitoring – Software Defined Networking – Network Function Virtualization – MapReduce – Identity and Access Management – Service Level Agreements – Billing.	**06**

II	**Cloud Services** **Compute Services:** Amazon Elastic Computer Cloud - Google Compute Engine - Windows Azure Virtual Machines. **Storage Services:** Amazon Simple Storage Service - Google Cloud Storage - Windows Azure Storage **Database Services:** Amazon Relational Data Store - Amazon Dynamo DB - Google Cloud SQL - Google Cloud Data Store - Windows Azure SQL Database - Windows Azure Table Service **Application Services:** Application Runtimes and Frameworks - Queuing Services - Email Services - Notifiction Services - Media Services **Content Delivery Services:** Amazon CloudFront - Windows Azure Content Delivery Network **Analytics Services:** Amazon Elastic MapReduce - Google MapReduce Service - Google BigQuery - Windows Azure HDInsight **Deployment and Management Services:** Amazon Elastic Beanstack - Amazon CloudFormation **Identity and Access Management Services:** Amazon Identiy and Access Management - Windows Azure Active Directory **Open Source Private Cloud Software:** CloudStack – Eucalyptus - OpenStack	**06**

III	**Cloud Application Design:** Introduction – Design Consideration for Cloud Applications – Scalability – Reliability and Availability – Security – Maintenance and Upgradation – Performance – Reference Architectures for Cloud Applications – Cloud Application Design Methodologies: Service Oriented Architecture (SOA), Cloud Component Model, IaaS, PaaS and SaaS Services for Cloud Applications, Model View Controller (MVC), RESTful Web Services – Data Storage Approaches: Relational Approach (SQL), Non-Relational Approach (NoSQL).	**06**
IV	**Cloud Application Benchmarking and Tuning:** Introduction to Benchmarking – Steps in Benchmarking – Workload Characteristics – Application Performance Metrics – Design Consideration for Benchmarking Methodology – Benchmarking Tools and Types of Tests – Deployment Prototyping. **Cloud Security:** Introduction – CSA Cloud Security Architecture – Authentication (SSO) – Authorization – Identity and Access Management – Data Security: Securing data at rest, securing data in motion – Key Management – Auditing.	**06**
V	**Case Studies:** Cloud Computing for Healthcare – Cloud Computing for Energy Systems - Cloud Computing for Transportation Systems - Cloud Computing for Manufacturing Industry - Cloud Computing for Education.	**06**

Extended Professional Component (is a part of internal component only, Not to be included in the External Examination question paper)	Questions related to the above topics, from various competitive examinations UPSC / TRB / NET / UGC – CSIR / GATE / TNPSC / others to be solved (To be discussed during the Tutorial hour)	
Skills acquired from the course	Knowledge, Problem Solving, Analytical ability, Professional Competency, Professional Communication and Transferrable Skill	

Learning Resources:

- **Recommended Texts**

1. Arshdeep Bahga, Vijay Madisetti, *Cloud Computing – A Hands On Approach*, Universities Press (India) Pvt. Ltd., 2018.

- **Reference Books**

1. Anthony T Velte, Toby J Velte, Robert Elsenpeter, *Cloud Computing: A Practical Approach*, Tata McGraw-Hill, 2013.
2. Barrie Sosinsky, *Cloud Computing Bible*, Wiley India Pvt. Ltd., 2013.
3. David Crookes, *Cloud Computing in Easy Steps*, Tata McGraw Hill, 2012.
4. Dr. Kumar Saurabh, *Cloud Computing*, Wiley India, Second Edition 2012.

Web resources: Web resources from NDL Library, E-content from open-source libraries