RATHNAVEL SUBRAMANIAM COLLEGE OF ARTS & SCIENCE (AUTONOMOUS) SULUR, COIMBATORE-641402

DEPARTMENT OF SCHOOL OF COMPUTER STUDIES

B.Sc., INFORMATION TECHNOLOGY

Syllabus effective for the students admitted during the Academic Year 2016-2017 Batch & onwards

(2016 - 2019)

HOD **PRINCIPAL** **COE**

SCHEME OF EXAMINATIONS

SCHOOL OF COMPUTER STUDIES (UG) - B.Sc. (IT)

2016 BATCH– BASED ON CHOICE BASED CREDIT SYSTEM

Semester	Part	Type	Title of the Paper	Hours of Instruction / week		Credits	Duration of Examination in Hours	Marks		
				Lecture Hours	Tutorial Hours			CIA	EOS	Total
I	I	L-I	Tamil/Arabic/French/Hindi /Malayalam – I	6	-	4	3	25	75	100
	II	E-I	English I- Grammar and Usage	5	1	4	3	25	75	100
	III	M-I	C Programming	6	-	4	3	25	75	100
	III	MP-I	Programming Lab I - (C Programming)	7		5	3	40	60	100
	III	A-I	Computer Oriented Numerical and Statistical Methods	5	-	4	3	25	75	100
	IV	FC-A	Foundation Course – A- Value Education Environmental Studies [Self Study]	-	-	2	3	50	-	50
	V	EC	NCC/NSS / Sports / Games	-	-	-	-	-	-	-
II	I	L-II	Tamil/Arabic/French/Hindi /Malayalam – II	6	-	4	3	25	75	100
	II	E-II	English II- Communicative English	5	1	4	3	25	75	100
	III	M-II	Object Oriented Programming using Java	5	-	4	3	25	75	100
	III	M-III	Object Oriented Design using UML	3	-	3	3	25	75	100
	III	MP-II	Programming Lab- II (Java Programming)	5	-	4	3	40	60	100
	III	A-II	Mathematical Foundation for Technology	5	-	4	3	25	75	100
	IV	FC-B	Foundation Course – B General Awareness[Self Study]	-	-	2	3	50	-	50
	V	EC	NCC/NSS / Sports / Games	-	-	-	-	-	-	-

Semester	Part	Type	Title of the Paper	Hours of Instruction / week		Credits	Duration of Examination in Hours	Marks		
				Lecture Hours	Tutorial Hours			CIA	EOS	Total
III	III	M-IV	Data Structures	4	1	4	3	25	75	100
	III	M-V	Software Engineering	5	-	4	3	25	75	100
	III	M-VI	Operating Systems	4	1	4	3	25	75	100
	III	M-VII	Fundamentals of Networks	5	-	3	3	25	75	100
	III	MP-III	Programming Lab – III (Data Structures & Fundamentals of Networks)	5(3+2)	-	4	3	40	60	100
	III	A-III	Operations Research for Information Technology	5	-	4	3	25	75	100
	V	EC	NCC/NSS / Sports/ Games	-	-	-	-	-	-	-
IV	III	M-VIII	Design and Analysis of Algorithms	4	1	4	3	25	75	100
	III	M-IX	Basic IP services	6	-	4	3	25	75	100
	III	MP-IV	Programming Lab – IV (Algorithms & Basic IP services)	6(3+3)	-	4	3	40	60	100
	III	A-IV	Business Accounting	5	-	4	3	25	75	100
	III	EL-I	Elective – I	5	-	4	3	25	75	100
	IV	SBC-I	Aptitude Skills	3	-	3	3	100	-	100
	V	EC	NCC/ NSS / Sports / Games	-	-	1	-	100	-	100

	Part	Type	Title of the Paper	Hours of Instruction / Week		Credits	Examination Hours	Marks		
								CIA	EOS	Total
V	III	M-X	Relational Database Management System	4	1	5	3	25	75	100
	III	M-XI	PHP & MySQL	6	-	4	3	25	75	100
	III	M-XII	AngularJS	6	-	4	3	25	75	100
	III	MP-V	Programming Lab – V (Relational Database Management System & PHP & MySQL)	6(3+3)	-	4	3	40	60	100
	III	EL-II	EDC – Cyber Security	3	-	3	3	25	75	100
	IV	SBC-II	Advanced IP services	4	-	3	3	25	75	100
VI	III	M-XIII	IPV-4 Routing	6	-	5	3	25	75	100
	III	M-XIV	Android programming	6	-	4	3	25	75	100
	III	MP-VI	Programming Lab-VI (MongoDB& IPV-4 ROUTING)	6	-	4	3	40	60	100
	III	MPV-I	Project & Viva Voce	7	-	5	3	40	60	100
	III	EL-III	Elective –III	4	1	4	3	25	75	100
			Total			**140**				**3600**
III-VI		CS	**Extra Credits:** (i) Career Skills (Mandatory)			3	-	-	-	Grade
V		ALCTA	(ii) Big Data (Self Study) (Optional)			4	-	-	-	Grade
I-IV		DM	(iii) Fire Fighting Techniques			1	-	-	-	-
		FA	(iv) First Aid Training			1	-	-	-	-

L – Language; **E** – English; **M** – Major Paper; **MP** – Major Practical; **A**– Allied paper;
EL – Elective; **EC** – Extra Curricular Activities; **SBC** – Skill Based Course; **EDC**– Extra Disciplinary Course; **ALCTA** – Advanced Learners Course in Thrust Area; **MPV**-Major Project Viva-Voce, **CS**- Career Skills; DM- Disasters Management; FA-First Aid**.**

Elective – I
a) Web Basics
b) Compiler Design

Elective- III
a) MongoDB
b) Enterprise Java Beans

SEMESTER - I
LANGUAGE-THEORY
FRENCH - I

OBJECTIVES :

On fait les etudiants parler couramment le francais. On fait les etudiants construire les phrases grammaticalement correctes. On fait les etudiants connaitre de la vie sociale, gracea l'acquisition de savoir et savoirs-faire communicatifs, linguistiques et culturels.

HOURS / WEEK - 6		
HOURS / SEMESTER - 90		
CIA	**CO-CURRICULAR ACTIVITIES**	**LECTURE**
12	6	72

UNIT 1 (LECTURE HOURS: 12)

Vous comprenez ?

Les Informations initiales et Parcours d'initiation - Les renseignements sur la France, le Francais, la langue francaise, reciter les alphabets, les nombres en francais et comment on les prononce. Comment vous vous appelez, Quel age avez-vous, Que faites-vous , Quelle est votre nationalite, Vous habitez ou, Qu'est-ce que c'est, Qui est-ce? **Conjugaison** - Classification des verbes, temps, formes et Comment conjuguer avec les pronoms sujets.
Masculin et Feminin - Distinguer les noms par Genre (Masculin et Feminin)
Singulier et Pluriel - Distinguer les noms par Nombre (Singulier et Pluriel) **Se presenter a un groupe** - Je m'appelle, C'est moi, etc., **Les sons et les lettres** - Les voyelles et les consonnes.

UNIT 2 (LECTURE HOURS: 12)

Au Travail !

Est-ce que vous connaissez ? - Nommer et preciser les pays, les gens et les choses.
Conjugaison - Conjuguer les verbes reguliers en (-er / -ir) present.
Les Noms et Les adjectifs - Accord des noms et des adjectifs selon genre et nombre.
Les Articles indefinis et definis - Indiquer les noms par les articles indefinis et definis.
Demander quelqu'un et quelque chose - Comment identifier une personne, un objet.
Enumerer et Exprimer - Enumerer ce que l'on connait, ce que l'on aime. Exprimer les gouts.

UNIT 3 (LECTURE HOURS: 12)

On se detend ?

Conjugaison - Conjuguer les verbes irreguliers au present eg:- faire, aller, venir, vouloir, pouvoir etc.,
Les Pronoms - Classification des pronoms et comment utiliser les pronoms (moi, toi, lui, elle, etc....)
Parler de quelque chose - Parler des loisirs, les activites differentes.
Proposer quelque chose - Accepter ou refuser une proposition.
Demander une explication - Exprimer la possibilite, l'impossibilite, l'obligation.
Cartes et Messages - Comment ecrire des cartes et des messages d'invitation, d'acceptation ou de refus.

UNIT 4 (LECTURE HOURS: 12)

Racontez-moi

Conjugaison - Conjuguer les verbes reguliers et irreguliers au Passe Compose.
Date et Heure - Comment ecrire la date et l'heure.
Evenements lies au temps - Decrire les moments de la journee, de l'annee.
Raconter un emploi du temps - Donner de precisions sur le temps, Differenciation entre Present et Passe.
Feliciter - Comment feliciter quelqu'un et quelque chose.
Journal Personnel - Ecrire des choses personnelles chaque jour dans le journal personnel.

UNIT 5 (LECTURE HOURS: 24)

Bon Voyage ! et Bon appetit !

Comparaison simple - Faire une comparaison simples des noms et des choses.

SEMESTER - I
LANGUAGE-THEORY
FRENCH - I

Adjectifs Demonstratifs et Possessifs - Indiquer les noms par des adjectifs demonstratifs et possessifs.
Voyages et Transports - Faire des voyages, Choisir des transports.
Presenter quelque chose - Presenter les avantages et les inconvenients d'une activite.
Faire des choses - Choisir et Negocier une activite commune. Demander et donner une explication.
Situer des choses - Expliquer les situations pratiques relatives au voyage.
Emploi des Articles - Indiquer et utiliser les noms par des articles partitifs.
Interrogation - Ecrire les phrases interrogatives avec l'inversion.
Reponses - Repondre les questions avec oui, si, non.
Nourriture, Repas et Fete - Decrire la nourriture, les repas et les fetes.
Situer des evenements - Expliquer des situations pratiques a l'hotel et au restaurant.
Transports et Habitudes - Expliquer les transports en France et les habitudes alimentatires des francais.

Text Books :
Echo – A1- Methode de francais | Edition:1 | La collaboration de C.GIBBE | J.GIRARDET et J.PECHEUR (2012)

SEMESTER - I
LANGUAGE-THEORY
HINDI - I

OBJECTIVES :

Rashrabhasha Hindi ke prati jagarooka utpanna karana HIndi ki mahanta aur uske vyavabarikta ke baare me cchatron ko avagat karana

HOURS / WEEK - 6		
HOURS / SEMESTER - 90		
CIA	**CO-CURRICULAR ACTIVITIES**	**LECTURE**
12	**6**	**72**

UNIT 1 (LECTURE HOURS: 30)

Kahaniyan

Khel - Jainendra kumar ka parichay, kahani ka saaramsh, kahani ka bhav, kahani ka uddehshay **Album** - lekhak ka parichay, kahani ka saar, kahani ka udheshay, kahani ka bhav

dukhwa kase kahoon mori sajni - , lekhak ka parichay, kahani ka saar, kahani ka udheshay, kahani ka bhavkhel kahani me nihith bhav.

vapasi - , lekhak ka parichay, kahani ka saar, kahani ka udheshay, kahani ka bhavkhel kahani me nihith bhav.

raju - , lekhak ka parichay, kahani ka saar, kahani ka udheshay, kahani ka bhavkhel kahani me nihith bhav.

UNIT 2 (LECTURE HOURS: 30)

Gadya

kadamb ka phool - lekhak parichay, gadya ka saar, gadya ka bhav, gadya me nihith tatva

gaura gai - gai ka arichay, karuna se bhara desh, ab dsh ki halath, janvaron ke prti logon ka badalta drisya **imaan daron ke sammelan me** - parisai ka parichay, immandar kai paribhasha, ajjkal ke immandaar log **shangai ki dairy** - shangai ki adbhudh drishya, china ka paridrishya, china me shiksha ka drishya

jeevan ki teen pradaan bhate - , lekhak ka parichay, kahani ka saar, kahani ka udheshay, kahani ka bhavkhel kahani me nihith bhav.

UNIT 3 (LECTURE HOURS: 4)

Laghu katha

laghu kathaye - paanch kahaniyan, uska tatva

UNIT 4 (LECTURE HOURS: 4)

Vyakaran

vyakaran - sangya, sarvanaam, karak, visheshan

UNIT 5 (LECTURE HOURS: 4)

Anuvad Abbyas

angreji se Hindi - das anuvad abbyas

Text Books :

vividh maala | Edition:1 | Dept of Hindi, RVS CAS | Dept of Hindi(2011)

SEMESTER - I
LANGUAGE-THEORY
MALAYALAM - I

OBJECTIVES :

To enable the students to learn and imbibe the values, culture and ethics through comprehensive studying and thereby appreciating the selected short stories and novel.

HOURS / WEEK - 6		
HOURS / SEMESTER - 90		
CIA	**CO-CURRICULAR ACTIVITIES**	**LECTURE**
12	6	72

UNIT 1 (LECTURE HOURS: 16)

Short story

Vellappokkathil - The legendary writer illustrated the selfish mentality of a man, who ignored his pet which was him for a pretty long period, during a natural disaster. The story recalls the importance of being kind to all flora and fauna around us.

Shabdhikkunnakalappa - The story is shear contradictory to the preceding one. A farmer named Ouseph who is deeply attached to an ox which assists him in his agriculture land. Finally both of them are closely associated and the ox becomes an important member of Ouseph's family.

Kulathozil - Sethu's father is a seasoned accountant who wants his son to select the same profession . To escape from his father's decision Sethu eloped from home and due to circumstances he returned home and finally selected his father' s desire by sacrificing his passion. However at the end of story Sethu will also become a renowned accountant.

UNIT 2 (LECTURE HOURS: 12)

Short story

Kadaltheerath - Vellayiappan hailing from a remote village wants to see his son in jail who has sentenced to death . But unfortunately he was able to see only the son's dead body and finally . The story demonstrates the mental status of such an unfortunate father.

Ammayum Makanum - Unni a boy who disliked and ignored the love of his mother due to her poor physical appearance and hence he don't want to introduce her to any of his circle. Finally the boy is deeply regretting and realizing her love after her death.

Mucheetukalikkarantey Makal - A comical story told by Basheer, the douyen of Malayalam stories. A satire where three characters behind the hunt of truth behind a petty theft.

UNIT-3 (LECTURE HOURS: 12)

Novel

Enmakaje - Ambika Sudhan was born in kasaragode. Professor in Nehru College, Mangad.Winner of lot of prestigious awards.A very touching novel and contemporary too . The novel describes the deleterious effects of use of a pesticide named 'endosulfan'

UNIT 4 (LECTURE HOURS: 18)

Novel

Enmakaje - Devayani and Neelakandan lives together in a deep forest avoiding the contacts of rest of world. One fine morning Devayani found an unusual baby in the deep forest and brings to Neelakandan but he rejected the due to its unusual morphology. But Devayani decided to nurse the child inspite of his objection.The situations are progressing in an unpredictable way. The sad part is that both of them strongly believe that the disease of the child is a result of curse from God. The mind set of both the characters is changed in the mid of the novel and they decided to come out of the forest of giving medical aid to the child.From the doctor they came to know that this disease is due to exposure of an endosulfan and it affects the animals around also.The ill effects of the pesticide are also scientifically challenged by the author in many crucial situations

UNIT 5 (LECTURE HOURS: 14)

Novel

SEMESTER - I
LANGUAGE-THEORY
MALAYALAM - I

Enmakaje - From the doctor they came to know that this disease is due to exposure of an endosulfan and it affects the animals around also. The medical team was unable to save the child. The issue attracted the nedia attention. Jayarajan , social worker who took the matter and fought with the Government for the tolal ban of endosulfan. Due to interference of some rich farmers bureaucracy has tried its maximum to avoid the total ban of this hazardous chemical. At the end of the novel that good news is mentioned because Govt. of Kerala has decided foe a total ban of this destructive pesticide.

Text Books :

Enmakaje | Edition:2 | DC books | Ambika Sudhan(2009)
vellapokkathil | Edition:4 | DC books | Thakazhi(1996)

Reference Books :

Cherukatha EnnaleyEnnu | Edition:6 | DCbooks | Sankarank p(1998)
Malayalasahithyam Kalaghattanghaliludey | Edition:5 | Current books | Parameswaran pillai(2003)

SEMESTER - I
ENGLISH-THEORY
ENGLISH I - GRAMMAR AND USAGE

OBJECTIVES :

To help the students to develop LSRW skills, improve their vocabulary, familiarize the different aspects of functional english and guide them to write effectively in informal and business situation .

HOURS / WEEK - 6		
HOURS / SEMESTER - 90		
CIA	**CO-CURRICULAR ACTIVITIES**	**LECTURE**
12	6	72

UNIT I (LECTURE HOURS: 14)

UNITS OF LANGUAGE

DEFINITIONS - A Letter is any one of the symbols used in the alphabet a ,b,c

Main Clause and Subordinate Clause - If a clause conveys complete meaning it is called a Main Clause If the meaning of a clause is incomplete, it is called a Subordinate Clause

EXERCISES - Identify the Main and the Subordinate clauses in the following sentences

KINDS OF SENTENCES

Definitions and kinds of Simple Sentences - An assertive sentence is a sentence in the form of a statement **Definitions and kinds of Non-Simple Sentences** - Compound Double sentence Compound Multiple sentence Complex sentence

Exercise - identify the kinds of sentences

PARTS OF SPEECH

Definition of Noun, Pronoun, Adjective - A noun is a word used as the name of a person, place, thing, state, quality etc

Definition of Verb, Adverb, Preposition, Conjunction, Interjection - Verb is a word that denotes and action or a state

Preposition is a word that is used with a suffix

Exercise - Name the parts of speech in the following sentences

NOUN: KINDS

DEFINITION of Proper Noun, Common Noun - A Proper Noun is the name given to a particular person, place or thing.

A Common Noun is the name given in common to persons and things .

Definition of Collective Noun, Abstract Noun - Collective Noun is the name given to a collection or group of persons or things taken as a single group. An Abstract Noun is the name given to a quality or action or state which we can understand but cannot see or touch.

Exercises - Identify the kinds of nouns in the given sentences

TUTORIAL 1

Quiz - Parts of Speech

TUTORIAL 2

Discussion - Nouns and its kinds

UNIT II (LECTURE HOURS: 16)

VERB

Transitive and Intransitive verbs - A transitive verb is a verb that takes an object

Auxiliary verbs, Defective Verbs - A Verb which helps another verb to form its Tense, Voice or Mood is called an Auxiliary Verb

Forms of Verb - Past present and past participle

Exercises - Identify the kinds of words

THE VERB "BE"

Be forms - Am, Is, Are, Was, Were

Do and Have - Do, Does Did, Done and Have, Has, Had

Exercise - Identify the role of Be, do and have

THE TENSE

Tenses - Are in active and passive voices

SEMESTER - I
ENGLISH-THEORY
ENGLISH I - GRAMMAR AND USAGE

Active voice - There are 12 tenses in active voice
Passive voice - There are 8 tenses in passive voice
PASSIVE VOICE
Definition - Passive is used when the doer of an action is unknown
Passive voice with tenses - Simple present tense, simple past tense and simple future tense.
Exercise - Change the following sentences into passive voice
TUTORIAL 1
Quiz - Verbs and forms
TUTORIAL 2
Discussions - Tenses in daily life
TUTORIAL 3
Activity - Games on transformation

UNIT III (LECTURE HOURS: 12)

NEGATIVE AND INTERROGATIVE SENTENCES
Negative sentences - Negative sentences are formed by adding "Not" after auxiliary verbs.
Interrogative Sentences - Interrogative Sentences are with "Be" verbs and with question words and without question words.
Exercises - Convert the following sentences into Negative sentences
TAG QUESTIONS
An introduction to Tag questions - Tag questions are used often in conversations and informal writing : They are short questions added to statements" asking for agreement or confirmation".
Positive and Negative tag questions - If the statement is positive, the tag will be negative, if the statement is negative the tag will be positive:
Exercise - Add appropriate tags to the following sentences
INFINTIVES AND GERUNDS
Definition to infinitives - An Infinitive can be used as the subject of a verb
Defitinition to Gerunds - A gerund is used as the subject of a verb
Exercise - Fill in the blanks with Gerund or the Infinitive form of the verbs
THE ARTICLES
Introduction to the Articles, Types of Articles - Three adjectives "a", "an" and "the" are given the special name of "articles". Indefinite articles and definite articles
Exercises - Fill in the blanks with articles
TUTORIAL 1
Quiz - Negative and Interrogative sentences

UNIT IV (LECTURE HOURS: 15)

AGREEMENT OF THE VERB WITH THE SUBJECT
Verb and Subject - The Verb in every sentence must agree with the Subject in Person and Number **Singular verbs** - Singular subjects are plural and singular noun must be singular **Plural verbs** - If there are two articles or possessives the verb must stand plural
Two are more singular subjects - Subjects connected by "OR", "NOR", "EITHER..OR", "EITHER..NOR" take a singular verb.
Verbs with first person and second person - If one subject is of the first person the verb is also first person plural. **Collective nouns, Exercise** - For collective noun a singular noun is used. In the following senttenses supply verbs in agreement with the subject
DIRECT AND REPORTED SPEECH
Introdution on Direct speech - Quoting the actual words spoken by a person is called Direct Speech
Introduction to reported speech - Reported speech is written in third person narrative.
Rules for changing direct to reported speech - If the Reporting Verb is in the past tense and the portion within the inverted commas is in the form of a statement, then the reported section is introduced after the word 'that'
Tenses and Questions - 'Present Tense' is changed into 'Past Tense' within the statement inverted commas. If the section within the inverted commas is in the form of a question, the word that is not used before the reported part
Command, Requests and Exclamations - Commands include ask, order, requests. Exclamatory sentences must be changed into statements forms.
Exercises - Change the following sentences into reported speech.
TUTORIAL 1

SEMESTER - I
ENGLISH-THEORY
ENGLISH I - GRAMMAR AND USAGE

Quiz - Agreement of the verb with the subject
TUTORIAL 2
Discussion - Direct and Reported speech
TUTORIAL 3
Activity - Convert Direct speech into reported speech

UNIT V (LECTURE HOURS: 15)

TRANSFORMATION OF SENTENCES - I

Introduction to Affirmative sentences into Negative sentences - The Antonym of an important word from a sentence is taken to form a negative affirmative sentence .
Negative sentences to Affirmative sentences - The synonym of an important word from a sentence is taken to form a Affirmative sentence.,
Exclamatory sentences to assertive sentences - To change exclamatory sentences into assertive sentences , we express their meanings and emotions in the statement form
Assretive to Exclamatory sentences - In exclamatory sentences the emotions are expressed in an arresting manner **"NO SOONER...THAN" and "As SOON AS" types, Scarcely... when** - In "NO SOONER types the verb is split into two parts, it is another form of expressing ideas in past perfect tense or the past tense.
"TOO...TO" and "SO..THAT..NOT" types, Exercise - If the "TOO...TO" patterns sentence is in the present tense we use "CAN". Change the following sentence patterns

PUNCTUATION

Definition on Punctuation - Punctuation marks are used for the better understanding of the sentences.
Full stop, Commas and Semi colon - A full stop indicates the completion of a sentence
Colon, Mark of Interrogation and Exclamation - Mark of Interrogation is used at the of every direct question **Dash,**
Inverted commas, Hyphen - Hyphen is used with compound words
Parenthesses, Capital letters - At the begining of all sentences Capital letters must be used.
Exercise I & II - Punctuate the following sentences, Use appropriate punctuation marks in the following sentences
TUTORIAL 1
Quiz - Transformation of sentences
TUTORIAL 2
Discussion - Punctuation and its importance
TUTORIAL 3
Activity - Conversion of Affirmative sentences into negative sentences

Text Books :
Know Your Grammar | Edition:1 | RVS Group of Institutions | Dr. Hyacinth Pink(2014)

Reference Books :
High School English Grammar & Composition | Edition:1 | S. Chand Publishers | Wren and Martin(2011)
Oxford Advance Learner's Dictionary | Edition:1 | Oxford University Press | A.S Hornby(2010)

SEMESTER - I
MAJOR-THEORY
C PROGRAMMING

OBJECTIVES :

- To inculcate knowledge on C Programming concepts .
- To understand the basic concepts of Structured Programming Model , Identifiers, Arrays, Control Statements, Functions, Pointers, Formatted I/O, Characters and Strings, Structure, Unions and File I/O.
- Enable the students to apply the techniques of functions to develop console applications.

HOURS / WEEK - 6		
HOURS / SEMESTER - 90		
CIA	**CO-CURRICULAR ACTIVITIES**	**LECTURE**
12	**6**	**72**

UNIT I (LECTURE HOURS: 12)

HARDWARE AND SOFTWARE-PROGRAMMING LANGUAGES-THE C PROGRAMMING LANGUAGE-TYPICAL C PROGRAM DEVELOPMENT ENVIRONMENT-A SIMPLE C PROGRAM: PRINTING A LINE OF TEXT - ANOTHER SIMPLE C PROGRAM: ADDING TWO INTEGERS

Moore's Law - computer programs,programmers,software,Moore's law
Computer organization - logical units of a computer(table)
machine languages,assembly languages,high level languages - machine,assembly and high level languages,interpreter **Built for Performance,Standardization,The new C standard** - portable,some popular performance oriented C applications(table),hardware platforms
Phase 1-6,problems that may occur at execution time,standard input-output and error streams - typical C development environment(diagram)
comments - comments, document programs, multi line comments, %...% multi line comments **#include preprocessor directive** - C preprocessor, standard input output header()
blank lines and white space - white space
the main function - function, body, left and right brace
an output statement - action, character string or message or literal, argument, statement, statement terminator **escape sequences** - escape character, escape sequence, new line, some common escape sequences(table) **using multiple printfs** - executable
variable and variable definitions - Definitions,variables,integer, Example code: addition program **identifiers and case sensitivity** - identifier, case sensitive
syntax error - placing variable definitions among executables statements cause syntax error **prompting messages** - prompt message using printf with new line
the scanf function and formatted inputs - scanf, format control string, %d conversion specifier, address operator, Enter key, Interactive computing
assignment Statement - assignment statement, operands
printing with a format control string - first argument: the format control string and second argument: the value to be printed **calculations in printf statements** - calculations can be performed inside printf statements

MEMORY CONCEPTS - ARITHMETIC IN C
- DECISION MAKING: EQUALITY AND RELATIONAL OPERATORS

Introduction: memory concepts - type, value, memory location showing the name and value of a variable, destructive, memory location after both variables are input, non destructive, memory location after a calculation
integer division and the remainder operator - arithmetic operators(table), integer division, remainder operator **arithmetic expressions in straight line form** - straight line form
parentheses for grouping sub expressions - Parentheses are used in the same manner in algebraic expressions
rules of operator precedence - four rules for operator precedence, associativity, precedence of arithmetic operators(table) **sample algebraic and C expressions** - expression calculates arithmetic mean(average), equation of a straight line, expression contains remainder(%)
evaluation of a second degree polynomial - multiplication, division, addition, subtraction and assignment operations on the same level parenthesis expression, Example Code: order in which a second degree polynomial is evaluated

SEMESTER - I
MAJOR-THEORY
C PROGRAMMING

Introduction: Decision Making - decisions, if statement, condition, equality and relational operators (table), Example code: using if statements, relational and equality operators , Precedence and associativity of the operators (table), C's keywords (table)

ALGORITHMS AND PSEUDO CODE - CONTROL STRUCTURES -
THE IF -
THE IF…ELSE - THE WHILE REPETITION STATEMENT
Selection Statements in C
Introduction: Algorithms and Pseudo code - Sequential execution, transfer of control, go to statement, control structures: sequence, selection and repetition structure
Introduction : Control Structures - special purpose symbols: rectangles, diamonds, rounded rectangles and small circles, Flow charting C's sequence structure(diagram), small circle symbol(connector), decision symbol(diamond)
Flow charts - single selection statement, double selection statement, multiple selection statement **Repetition Statements in C** - single entry / single exit control statements, control statement stacking
pseudo code statement of if selection statement - if statement written in c, flow charting the single selection if statement **pseudo code statement of if…else and nested if …else selection statements** - if … else statement written in c-flow charting the double selection if …else statement, Nested if statement written in c
pseudo code statement of while repetition statement - while statement written in c, flow charting the while repetition statement **case study 1: Problem statement** - Pseudo code algorithm that uses counter controlled repetition to solve the class-average problem, Example code: Class average program with counter, controlled repetition
case study 2: Problem statement, Top Down-Stepwise Refinement, Second Refinement, Converting Between Types Explicitly and Implicitly, Formatting Floating Point Numbers, Notes on Floating-Point Numbers - Pseudo code algorithm that uses sentinel, controlled repetition to solve the class average problem, Example code: Class average program with sentinel, controlled repetition
case study 3: Problem statement - Pseudo code for examination results problem, Example code: Analysis of examination results
ASSIGNMENT OPERATORS -INCREMENT AND DECREMENT OPERATORS
Several Assignment Operators - Additional assignment operator - Arithmetic assignment operators (table).
Unary Increment and Decrement Operators - increment operator, decrement operator , pre increment or pre decrement and post increment or post decrement, Increment and Decrement Operators (table), Example code: pre incrementing and post incrementing, Precedence and Associativity of the operators(table)

UNIT II (LECTURE HOURS: 12)

REPETITION ESSENTIALS - COUNTER CONTROLLED REPETITION
Counter controlled repetition - definite repetition, control variable **Sentinel controlled repetition** - sentinel values are used to control repetition

Counter Controlled Repetition requirements - name, initial value, increment (or decrement), final value, Example code: Counter Controlled Repetition
FOR REPETITION STATEMENT - FOR STATEMENT: NOTES AND OBSERVATIONS - EXAMPLES USING FOR STATEMENT - SWITCH MULTIPLE SELECTION STATEMENT -DO…WHILE REPETITION STATEMENT - BREAK AND CONTINUE STATEMENT

for Statement Header Components - Example code: Counter controlled repetition with the for statement, for Statement Header Components (diagram)
Off-By-One Errors - Off-by-one-error
General Format of a for Statement - General format of for and while statements **Comma-Separated Lists of Expressions** - Comma operators
Expressions in the for Statement's Header Are Optional - three expressions in the for loop are optional **Increment Expression Acts Like a Standalone Statement** - Example code
Introduction: for statement -notes and observation - Flowcharting a typical for repetition statement
Application: Summing the Even Integers from 2 to 100 - Example code: Summing the Even Integers from 2 to 100
Application: Compound-Interest Calculations - Example code: Compound-Interest Calculations
A Caution about Using Type float or double for Monetary Amounts - A Caution about Using Type float or double for Monetary Amounts
Formatting Numeric Output - Formatting Numeric Output
Reading Character Input - Example code: Counting letter grades with switch, ASCII Character set
Entering the EOF Indicator - z
switch Statement Details - controlling expressions-case labels
switch Statement Flowchart - Flowcharting a Switch multiple selection Statement with breaks **Ignoring Newline-Tab - Blank Characters in Input** - skip Newline-Tab - Blank Characters **Constant Integral Expressions** - test Constant Integral Expressions

SEMESTER - I
MAJOR-THEORY
C PROGRAMMING

Notes on Integral Types - several data types to represent integers
do…while Statement Flowchart - Example code: Using the do…while repetition statement, Flowcharting the do…while Statement
break Statement, continue Statement - Example code: Using the break statement in a for statement, Example code: Using the continue statement in a for statement.
LOGICAL OPERATORS - CONFUSING EQUALITY (==) AND ASSIGNMENT (=) OPERATORS - STRUCTURED PROGRAMMING SUMMARY

Logical AND (&&) Operator - Truth table for the logical AND (&&) operator **Logical OR (** - Truth table for the logical OR (
) **Operator** -) operator
logical negation (!) operator - Truth table for operator ! (logical negation)
Summary of Operator Precedence and Associativity - Summary of Operator Precedence and Associativity(table)
The _Bool Data Type - boolean type-true and false
lvalues and rvalues - lvalue (left values)-rvalue(right value)
Confusing == and = in Standalone Statements - Example C statements
Summary of sequence, selection and repetition statements - single-entry/single-exit sequence, selection and repetition statements, Rules for forming structured programs, Simplest flowchart, Repeatedly applying Rule 2 to the simplest flowchart, Applying Rule 3 to the simplest flowchart, Stacked, nested and overlapped building blocks, An unstructured flowchart.

PROGRAM MODULES IN C - MATH LIBRARY FUNCTIONS - FUNCTIONS-FUNCTION DEFINITIONS- FUNCTION PROTOTYPES

Introduction: Program Modules in C - Functions, C standard library, programmer defined functions, calling function or caller, called function, Hierarchical boss-function /worker function relationship(diagram)
Introduction: Math Library Functions - argument, Commonly used math library functions(table)
Introduction: Functions - local variables, parameters, software reusability, abstraction
main's return Type - Example code: Creating and using a programmer, defined function, format of function definition
Function maximum - Example code: Finding the maximum of three integers
Compilation Errors - compilation
Argument Coercion and "Usual Arithmetic Conversion Rules" - Arithmetic data types and their conversion specifications(table)

FUNCTION CALL STACK AND STACK FRAMES - HEADERS - PASSING ARGUMENTS BY VALUE AND BY REFERENCE-RANDOM NUMBER GENERATION
Function Call Stack in Action - tack, pushing, popping, last-in, first-out (LIFO) data structures, function call stack/program execution stack, stack frame, stack overflow, Example code: Demonstrating the function call stack and stack frames using a function square, Function call stack after the operating system invokes main to execute the program(diagram), Function call stack after main invokes square to perform the calculation(diagram), Function call stack after function square returns to main(diagram) **Introduction: Headers** - header, Some of the standard library headers(table)
Introduction: Passing Arguments by Value and By Reference - pass-by-value, pass-by-reference, side effects
Rolling a Six-Sided Die - Example code: Shifted, scaled random integers produced by 1 + rand() % 6
Rolling a Six-Sided Die 6,000,000 Times - Example code: Rolling a Six-Sided Die 6,000,000 Times
Randomizing the Random Number Generator - pseudo random numbers, randomizing-seeds, Example code: Randomizing the die-rolling program
Generalized Scaling and Shifting of Random Numbers - shifting value-scaling factor
Case study: A Game of Chance - Problem Statement, Enumeration, Game Ends on First Roll, Game Ends on a Subsequent
Roll, Control Architecture - Example code: Simulating the game of craps
STORAGE CLASSES-SCOPE RULES-
RECURSION-EXAMPLE USING RECURSION: FIBONACCI SERIES-RECURSION VS ITERATION
Local Variables - Storage class specifiers: auto, register, extern and static, storage duration, linkage, automatic storage duration, static storage duration, automatic variables
Static Storage Class - extern and static, several types of identifiers with static storage duration
Introduction: Scope rules - scope of an identifiers, function scope, information hiding, principle of least privilege, file scope, block scope, function-prototype scope, Example code: Scoping
Recursively Calculating Factorials - base case(s), recursive call or recursion step, recursive evaluation of 5!(diagram), Example code: Recursive factorial function
Order of Evaluation of Operands - golden ratio or golden mean, base case-n, Example code: Recursive fibonacci function, Set of recursive calls for fibonacci(diagram)
Exponential Complexity - Doubling effect, exponential complexity
Comparison: Recursion Vs Iteration - Control structure, Repetition structure and selection structure , repetition: repetition statement and repeated function calls, termination test, gradually approach termination, infinite recursion

SEMESTER - I
MAJOR-THEORY
C PROGRAMMING

UNIT III (LECTURE HOURS: 12)

ARRAYS AND DEFINING ARRAYS- ARRAY EXAMPLES- PASSING ARRAYS TO FUNCTIONS- SORTING ARRAYS- CASE STUDY: COMPUTING MEAN, MEDIAN AND MODE USING ARRAYS- SEARCHING ARRAYS

Introduction: Arrays and Defining Arrays - position number, zeroth element, 12-element array(diagram), subscript, name, value, Operator precedence and associativity(table) , Example C statements

Defining an Array and Using a Loop to Initialize the Array's Elements - Size_t, Example code: Initializing the elements of an array to zeros

Initializing an Array in a Definition with an Initializer List - Array initializers, Example code: Initializing the elements of an array with an initializer list

Specifying an Array's Size with a Symbolic Constant and Initializing Array Elements with Calculations - Symbolic Constant , replacement text, scalable, Example code: Initialize the elements of array s to the even integers from 2 to 20 **Summing the Elements of an Array** - Example code: Computing the sum of the elements of an array

Using Arrays to Summarize Survey Results - problem statement, Example code: Analyzing a student poll

Graphing Array Element Values with Histograms - Example code: Displaying a histogram

Rolling a Die 6,000,000 Times and Summarizing the Results in an Array - Roll a six-sided die 6,000,000 times

Using Character Arrays to Store and Manipulate Strings - null character, Example code: Treating character arrays as strings **Static Local Arrays and Automatic Local Arrays** - Example code: Static arrays are initialized to zero if not explicitly initialized. **Difference Between Passing an Entire Array and Passing an Array Element** - %p conversion specifier, scalars, const, Example code: Array name is the same as the address of the array's first element, Example code: Passing arrays and individual array elements to functions

Using the const Qualifier with Array Parameters - constant or const, Example code: Using the const type qualifier with arrays

Sorting Array values - Bubble sort or sinking sort, Example code: Sorting an array's values into ascending order

Case Study: mean, median, mode - Example code: Survey data analysis with arrays:- computing the mean, median and mode of the data

Searching an Array with Linear Search - key value, searching, linear search, binary search, search key, Example code: Linear search of an array

Searching an Array with Binary Search - Small or unsorted arrays, one-half, middle, second-half, Example code: Binary search of a sorted array

MULTIDIMENSIONAL ARRAYS - VARIABLE LENGTH ARRAYS

Two-Dimensional Array Manipulations - multi dimensional arrays, tables, double subscripted array, m by n array, Double-subscripted array with three rows and four columns (diagram), Example code: Initializing multidimensional arrays, Example code: Double-subscripted array manipulations.

Definition: Variable Length Arrays - Variable Length Arrays (VLA), Example code: Using variable-length arrays in C99

POINTER VARIABLE DEFINITIONS AND INITIALIZATION - POINTER OPERATORS - PASSING ARGUMENTS TO FUNCTIONS BY REFERENCE - USING THE CONST QUALIFIER WITH POINTERS - BUBBLE SORT USING CALL-BY-REFERENCE - SIZEOF OPERATOR - POINTER EXPRESSIONS AND POINTER ARITHMETIC

Declaring Pointers - indirection, Directly and indirectly referencing a variable(diagram)

Initializing and Assigning Values to Pointers - NULL is a symbolic constant

Pointer Representation in Memory - address operator, Graphical representation of a pointer pointing to an integer variable in memory (diagram)

The Indirection (*) Operator - Representation of y and yPtr in memory (diagram), Indirection Operator or dereferencing operator

Demonstrating the & and * Operators - Example code: Using the & and * pointer operators, Operator precedence and associativity(table)

Pass-By-Value - Cube a variable using pass-by-value **Pass-By-Reference** - Cube a variable using pass-by-reference with a pointer argument

Introduction - Analysis of a typical pass-by-value (diagram), Analysis of a typical pass-by- reference with a pointer argument (diagram)

Converting a String to Uppercase Using a Non-Constant Pointer to Non-Constant Data - Example code: Converting a String to Uppercase Using a Non-Constant Pointer to Non-Constant Data

Printing a String One Character at a Time Using a Non-Constant Pointer to Constant Data - Example code: Printing a String One Character at a Time Using a Non-Constant Pointer to Constant Data, Example code: Attempting to modify data through a non-constant pointer to constant data

Attempting to Modify a Constant Pointer to Non-Constant Data - Example code: Attempting to Modify a Constant Pointer to Non-Constant Data

Attempting to Modify a Constant Pointer to Constant Data - Example code: Attempting to Modify a Constant Pointer to Constant Data

SEMESTER - I
MAJOR-THEORY
C PROGRAMMING

Introduction: Bubble Sort Using Call-by-Reference - information hiding, software reusability and software engineering, Example code: Putting values into an array, sorting the values into ascending order and printing the resulting array

Determining the Sizes of the Standard Types, an Array and a Pointer - sizeof, Example code: Applying sizeof to an array name returns the number of bytes in the array, Example code: Using operator sizeof to determine standard data type sizes **Introduction: Pointer Expressions and Pointer Arithmetic** - Array v and a pointer variable vPtr that points to v (diagram), The pointer vPtr after pointer arithmetic (diagram), pointer to void(void *)

RELATIONSHIP BETWEEN POINTERS AND ARRAYS - ARRAYS OF POINTERS

String Copying with Arrays and Pointers - offset, pointer/offset notation, pointer/subscript notation, pointer subscripting, Example code: Using subscripting and pointer notations with arrays, Example code: Copying a string using array notation and pointer notation **Definition: Arrays of Pointers** - Arrays of Pointers array of string or string array, Graphical representation of the suit array (diagram).

Case Study: Card Shuffling and Dealing Simulation - Double-subscripted array representation of a deck of cards (diagram), Example code: Card shuffling and dealing

Pointers to Functions

Using Function Pointers to Create a Menu-Driven System - Pointers to a Function, passed to functions, returned from functions, stored in arrays, assigned to other function pointers, Example code: Multipurpose sorting program using function pointers program, Example code: Demonstrating an array of pointers to functions

UNIT IV (LECTURE HOURS: 12)

FUNDAMENTALS OF STRINGS AND CHARACTERS- CHARACTER HANDLING LIBRARY

Introduction: Fundamentals of Strings and Characters - Character constants, character set, special characters, string literals or string constant, null character, string is a pointer

Functions isdigit, isalpha, isalnum and isxdigit - Character-handling library () functions (table), Example code: Using functions isdigit-isalpha- isalnum and isxdigit

Functions islower- isupper-tolower and toupper - Example code: Using functions islower-isupper- tolower and toupper

Functions isspace-iscntrl-ispunct-isprint and isgraph - Example code: Using Functions isspace-iscntrl-ispunct-isprint and isgraph

STRING-CONVERSION, STANDARD INPUT/OUTPUT AND STRING-MANIPULATION LIBRARY CHARACTER AND STRING FUNCTIONS

Function strtod - String-conversion functions of the general utilities library(table), Example code: Using function strtod

Function strtol - Example code: Using function strtol

Function strtoul - Example code: Using function strtoul

Functions fgets and putchar - Standard input/output library character and string functions (table), Example code: Using Functions fgets and putchar

Function getchar - Example code: Using Function getchar

Functions strcat and strncat

Function sprint - Example code: Using Function sprint

Function sscanf - Example code: Using Function sscanf

Functions strcpy and strncpy - String-manipulation functions of the string-handling library(table), Example code: Using functions strcpy and strncpy Example code: Using Functions strcat and strncat

COMPARISON, SEARCH AND MEMORY FUNCTIONS OF THE STRING-HANDLING LIBRARY - OTHER FUNCTIONS

Function strchr
Function strcspn
Function strpbrk
Function strrchr
Function strspn
Function strstr
Function strtok

Introduction: Comparison Functions of the String-Handling Library - Search functions of the string-handling library(table), Example code: Using function strchr, Example code: Using function strcspn, Example code: Using function strpbrk, Example code: Using function strrchr, Example code: Using function strspn, Example code: Using function strstr, Example code: Using function strtok

Function memchr
Function memset
Other Functions of the String-Handling Library: Function strerror
function strlen

SEMESTER - I
MAJOR-THEORY
C PROGRAMMING

Function memcpy - Memory functions of the string-handling library(table), Example code: Using function memcpy, Example code: Using function memmove, Example code: Using function memcmp, Example code: Using function memchr, Example code: Using function memset

Function memmove - Example code: Using Function strerror

Function memcmp - Example code: Using function strlen

STREAMS AND FORMATTING OUTPUT WITH PRINTF - PRINTING INTEGERS, FLOATING POINT NUMBERS, STRINGS AND CHARACTERS - OTHER CONVERSION SPECIFIERS - PRINTING WITH FIELD WIDTHS AND PRECISION - USING FLAGS IN THE PRINTF FORMAT CONTROL STRING - PRINTING LITERALS AND ESCAPE SEQUENCES - READING FORMATTED INPUT WITH SCANF.

Introduction: Streams - standard input stream, standard output stream, standard error stream

Output Formatting with printf - format control string , conversion specifiers, flags, field widths, precisions and literal characters, conversion specifications, seven format capabilities

Introduction: Printing Integers - Integer conversion specifiers(table), Example code: Using the integer conversion specifiers **Introduction: Printing Floating Point Numbers** - Floating-point conversion specifiers(table), Example code: Using Floating-point conversion specifiers

Introduction: Printing Strings and Characters - Conversion specifier c-Conversion specifier s, Example code: Using the character and string conversion specifiers

Other Conversion Specifiers - conversion specifiers p and %, Example code: Using the p and % conversion specifiers

Introduction: Printing with Field Widths and Precision - Example code: Right justifying integers in a field, Example code: Printing integers, floating-point numbers and strings with precisions

Introduction: Using Flags in the printf Format Control String - Format control string flags(table), Example code: Right justifying and left justifying values, Example code: Printing positive and negative numbers with and without the + flag, Example code: Using the space flag, Example code: Using the # flag with conversion specifiers

Introduction: Printing Literals and Escape Sequences - Escape sequences(table)

Introduction: Reading Formatted Input with scanf. - Inputting all types of data, Inputting specific characters from an input stream and Skipping specific characters in the input stream, Conversion specifiers for scanf(table), Example code: Reading input with integer conversion specifiers, Example code: Reading input with floating-point conversion specifiers, Example code: Reading characters and strings, Example code: Using a scan set, inputting data with a field width, Example code: Reading and discarding characters from the input stream

UNIT V (LECTURE HOURS: 12)

STRUCTURE DEFINITIONS- INITIALIZING AND ACCESSING STRUCTURE MEMBERS- USING STRUCTURES WITH FUNCTIONS- TYPEDEF

Self-Referential Structures - derived data types, struct, structure tag, structure type and members, struct employee2- Self-Referential Structures

Defining variables of Structure Types - struct card

Structure Tag Names - Structure Tag Name is optional

Operations that can be performed on structures - assigning structure variables to structure to structure variables of the same type, taking the address(&) of a structure variable, accessing the members of a structure variable and using the sizeof opearator to determine the size of a structure variable

Introduction: Initializing Structures - struct card aCard = { "Three", "Hearts" };

Introduction: Accessing Structure Members - structure member operator(.) and the structure pointer operator (->), Example code: Structure member operator and structure pointer operator

Introduction: Using Structures with Functions - Structures may be passed to functions by passing individual structure members, by passing an entire structure or by passing a pointer to a structure. **Introduction: typedef** - typedef struct card Card

Example: High-Performance Card Shuffling and Dealing Simulation - Example code: Card shuffling and dealing program using structures

UNIONS

Union Declarations - union, share same space storage, union definition

Operations That Can Be Performed on Unions - four operations can be performed on unions

Initializing Unions in Declarations - Union may be initialized with a value of the same type as the first union member **Demonstrating Unions** - Example code: Displaying the value of a union in both member data types

BITWISE OPERATORS - BIT FIELDS - ENUMERATION CONSTANTS

Using the Bitwise AND-Inclusive OR-Exclusive OR and Complement Operators

Displaying an Unsigned Integer in Bits - bitwise opearators(table), Example code: Displaying an Unsigned Integer in Bits, results of combining two bits with the bitwise AND opearator &, Example code: Displaying an unsigned int in bits

SEMESTER - I
MAJOR-THEORY
C PROGRAMMING

Making Function display Bits More Scalable and Portable - the symbolic constant CHAR_BIT

Making Function display Bits More Scalable and Portable - Example code: Using the Bitwise AND-Inclusive OR-Exclusive OR and Complement Operators, Results of combining two bits with the bitwise inclusive OR operator

Making Function display Bits More Scalable and Portable - (table) , Results of combining two bits with the bitwise exclusive OR operator ^(table)

Using the Bitwise Left- and Right-Shift Operators - Example code: Using the Bitwise Left- and Right-Shift Operators

Bitwise Assignment Operators - Example code: Using the bitwise shift operators, Bitwise Assignment Operators(table)

Introduction: Bit Fields - bit field, member name, width, unnamed bit field, padding, unnamed bit field with a zero width, Example code: Representing cards with bit fields in a struct

Introduction: Enumeration Constants - enumeration constants, Example code: Using an enumeration

FILES AND STREAMS

Introduction: Files and Streams - end-of-file marker, stream, standard input-standard output-standard error-file descriptor, open file table, file control block (FCB), stdin-stdout-stderr,fgetc( )-fputc()-fgets()- fputs(), fscanf()-fprintf(), fread()- fwrite().

CREATING A SEQUENTIAL ACCESS FILE - READING DATA FROM A SEQUENTIAL: ACCESS FILE||RANDOM ACCESS FILES - CREATING A RANDOM ACCESS FILE - WRITING DATA RANDOMLY TO A RANDOM ACCESS FILE - READING DATA FROM A RANDOM-ACCESS FILE

Introduction: Creating a Sequential Access File - Example code: Creating a Sequential-Access File, End-of-file key combinations for various popular operating systems(table), Relationship between FILE pointers, FILE structures and FCBs(diagram), File opening modes(table)

Resetting the File Position Pointer - Example code: Reading and printing a sequential file, file position pointer, file offset

Credit Inquiry Program - Example code: Credit Inquiry Program, formatted input / output model

Introduction: Random-Access Files - C's view of a random-access file(diagram)

Creating a Random-Access File:problem statement - Example code: Creating a random-access file sequentially

Introduction: Writing Data Randomly to a Random Access File - Example code: Writing data randomly to a random-access file, File position pointer indicating an offset of 5 bytes from the beginning of the file(diagram)

Introduction: Reading Data from a Random-Access File - Example code: Reading a random-access file sequentially

#INCLUDE PREPROCESSOR DIRECTIVE -
#DEFINE PREPROCESSOR DIRECTIVE: SYMBOLIC CONSTANTS - #DEFINE PREPROCESSOR DIRECTIVE:MACROS - CONDITIONAL COMPILATION

Introduction: #include preprocessor directive - C preprocessor, symbolic constants and macros, conditional compilation, conditional execution of preprocessor directive-#include preprocessor directive, standard library headers **Introduction: #define preprocessor directive: symbolic constants** - #define directive, replacement text

Introduction: #define preprocessor directive:macros - macro identifier, replacement text, arguments, macro with arguments, expanded, back slash

Introduction: conditional compilation - conditional compilation, #if - #endif, #ifdef - ifndef, #elif, debuggers

#ERROR AND #PRAGMA PREPROCESSOR DIRECTIVE - # AND ## OPEARATORS - LINE NUMBERS -PREDEFINED SYMBOLIC CONSTANTS - ASSERTIONS- USING COMMAND LINE ARGUMENTS

Using #error and #pragma preprocessor directive - Example C statements **Using # and ## opearators** - Example C statements

Using Line numbers - #line preprocessor directive, Example C statements

Using Predefined symbolic constants - Predefined symbolic constants, some Predefined symbolic constants(table)

Using Assertions - assert, , abort, Example C statements

Using command line arguments - pass arguments to main( ) , argc and argv parameters, Example code: Using command line arguments

UNIT I - TUTORIAL (LECTURE HOURS: 2)

Tutorial

GCD Algorithm - Examples

UNIT II - TUTORIAL (LECTURE HOURS: 2)

Tutorial

Functions Revisit - Discussion on solving problems using functions

UNIT III - TUTORIAL (LECTURE HOURS: 3)

Tutorial

Copying sub array into another array - Problem definition and Coding

UNIT IV - TUTORIAL (LECTURE HOURS: 2)

SEMESTER - I
MAJOR-THEORY
C PROGRAMMING

Tutorial

Multidimensional Arrays and Pointers Revisit - Examples

UNIT V - TUTORIAL (LECTURE HOURS: 3)

Tutorial

Dynamic Memory Allocation(DMA) - malloc() and realloc()

Text Books :

T1.C How to Program | Edition:7th | Pearson Education | Harvey Deitel AND Paul Deitel(2013)

Reference Books :

R1.The C Programming Language | Edition:2nd | Prentice Hall Software Series | Brian W.Kernighan AND Dennis M.Ritchie(1988)
R2.Programming in C | Edition:2nd | Pearson Education | Ashok N.Kamthane(2011)
Website1:https://onlinecourses.nptel.ac.in/noc15_cs15/preview | Edition:- | NPTEL | Dr. Satyadev Nandakumar AND Prof. Amey Karkare(-
)
Website2:http://www.geeksforgeeks.org/c | Edition:- | - | --(-)

SEMESTER - I
MAJOR-PRACTICAL
PROGRAMMING LAB - I (C PROGRAMMING)

OBJECTIVES :

·Enable the students to learn the basic functions, principles and programming techniques of C language.
·To present the syntax and semantics of the C language as well as data types offered by the language .
·To understand the basic concepts of problem solving approaches and develop optimal program structure using conditional and iterative control structures and functions.
·Apply the techniques of structured (functional) decomposition to break a program into smaller pieces and describe the mechanics of parameter passing.

HOURS / WEEK - 7		
HOURS / SEMESTER - 105		
CIA	**CO-CURRICULAR ACTIVITIES**	**LECTURE**
14	**7**	**84**

UNIT I (LECTURE HOURS: 12)

Pythagorean Triples
Problem Statement - algorithm and coding
Palindrome Tester
Problem Statement - algorithm and coding
Finding a Factorial with equations
Problem Statement - algorithm and coding
Finding a Gas Mileage Program
Problem Statement - algorithm and coding
Sales Commission Calculator
Problem Statement - algorithm and coding

UNIT II (LECTURE HOURS: 12)

Recursive Greatest Common Divisor
Problem Statement - algorithm and coding
Towers of Hanoi Problem
Problem Statement - algorithm and coding
Triangular Matrix
Problem Statement - algorithm and coding
Sum of adjacent pairs
Problem Statement - algorithm and coding

UNIT III (LECTURE HOURS: 12)

Mean, Median and Mode Program with Modifications
Problem Statement - algorithm and coding
Card Shuffling and Dealing with Modification
Problem Statement - algorithm and coding

UNIT IV (LECTURE HOURS: 24)

Write Your Own String Copy and Concatenation Functions
Problem Statement - algorithm and coding
Writing the Word Equivalent of a Check Amount
Problem Statement - algorithm and coding

UNIT V (LECTURE HOURS: 24)

File Matching Program
Problem Statement - algorithm and coding
Telephone Number Word Generator
Problem Statement - algorithm and coding

SEMESTER - I
MAJOR-PRACTICAL
PROGRAMMING LAB - I (C PROGRAMMING)

Text Books :

T1.C How to Program | Edition:7th | Pearson Education | HarveyDeitel AND Paul Deitel(2012)

SEMESTER - I
ALLIED-THEORY
COMPUTER ORIENTED NUMERICAL AND STATISTICAL METHODS

OBJECTIVES :

On sucessful completion of this course the students should gain knowledge about solving the non linear equations,ordinary differential equations and correlation

HOURS / WEEK - 5		
HOURS / SEMESTER - 75		
CIA	**CO-CURRICULAR ACTIVITIES**	**LECTURE**
10	**5**	**60**

UNIT I (LECTURE HOURS: 12)

Curve fitting

Fitting a straight line - Formula and problems

Parabola by the method of Least squares - Formula and problems

The solution of numerical, algebraic and transcendental equations

Bolzano's method - Formula and problems

The method false position - Formula and problems

Newton's Raphson method - Formula and problems

Horner's method - Formula and problems

Solution of simultaneous linear algebraic equations

Gauss elimination method - Formula and problems

Gauss Jordan elimination method - Formula and problems

Gauss Seidal method - Formula and problems

UNIT II (LECTURE HOURS: 12)

Interpolation for equal intervals

Newton forward interpolation - Formula and problems

Newton backward interpolation - Formula and problems

Equidistant terms with one or more missing values - Formula and problems

Interpolation for unequal intervals

Lagrange's interpolation formula - Formula and problems

Numerical differentiation and integration

Newton's forward difference formula - Formula and problems

Newton's backward difference formula - Formula and problems

Trapzoidal rule - Formula and problems

Romberg's rule - Formula and problems

Simpson's one third rule - Formula and problems

Simpson's three eighth rule - Formula and problems

UNIT III (LECTURE HOURS: 12)

Numerical solutions of ordinary differential equations

Solution by Taylor's series First order differential equation - Formula and problems **Taylor's series second order differential equation** - Formula and problems

Rungekutta methods - Formula and problems

Second order Rungekutta method - Formula and problems

Forth order Rungekutta method - Formula and problems

Eigen values and Eigen vectors

Eigen values and Eigen vectors by using Power method - problems

UNIT IV (LECTURE HOURS: 12)

Theory of probability

SEMESTER - I

ALLIED-THEORY

COMPUTER ORIENTED NUMERICAL AND STATISTICAL METHODS

Mathematical Probability - simple problems
Statistical Probability - simple problems
Addition law - Theorem
Multiplication law - Theorem
Correlation
Types of Correlation - simple problems
Correlation coefficient - simple problems
Rank Correlation - simple problems

UNIT V (LECTURE HOURS: 12)

Theoretical distribution
Binomial distribution - Definitions and problems
Poisson distribution - Definitions and problems
Normal distribution - Definitions and problems
Linear regression analysis
Lines of Regression of X on Y and Y on X - simple problems
Coefficient of regression - simple problems

Text Books :

T1.Numerical Methods | Edition:1 Edition | S.Chand & Co | P.K.KANDASAMY,K.THILAGAVATHY,& K.GUNAVATHY.(1997)
T2.Fundamental of Statistics | Edition:11 Edition | SULTAN CHAND & CO | S.C.GUPTA & V.K.KAPPOR.(2002)

Reference Books :

R1.Statistical methods | Edition:37 Edition | SULTAN CHAND & CO | S.P.GUPTA.(2001)

SEMESTER - II
LANGUAGE-THEORY
FRENCH II

OBJECTIVES :

On fait les etudiants parler courramment le francais. On fait les etudiants construire les phrases grammaticalement correctes. On fait les etudiants connaitre de la vie sociale, grace a l'acquisition de savoir et savoirs-faire communicatifs, linguistiques et culturels.

HOURS / WEEK - 6		
HOURS / SEMESTER - 90		
CIA	**CO-CURRICULAR ACTIVITIES**	**LECTURE**
12	6	72

UNIT-1 (LECTURE HOURS: 12)

Quelle journée !

La conjugaison pronominale - Comment indiquer les actions personnelles et conjuguer les verbes prnominaux.

L'impératif - Comment indiquer les conseils ou les ordres et conjuguer les verbes à l'impératif.

Raconter sa journée - Comment raconter sa journée et les activités quotidiennes en utilisant les verbes pronminaux. **Demander des nouvelles** - Comment demander des nouvelles de quelqu'un et comment choisir, acheter, payer un objet, et comment s'informer sur la présence ou l'existence d'une personne ou d'un objet.

Extrait d'un guide - Comment faire un extrait d'un guide touristique : les activités gratuites en France.

Rédaction d'un document - Comment écrire une rédaction d'un bref document d'information et le comportment en matière d'achat et d'argent.

UNIT-2 (LECTURE HOURS: 12)

Qu'on est bien ici !

Prépositions et Adverbes de lieu - Comment utiliser les prépositions et les adverbes de lieu et les verbes exprimant un déplacement (emploi des prépositions).

Parler d'un cadre de vie - Comment parler d'un cadre de vie et décrire le logement, la localisation, l'orientation, l'état physique et le temps qu'il fait.

S'informer sur un itinéraire - Comment s'informer sur un itinéraire, une orientation et demander de l'aide, exprimer une interdiction.

Différenciation des adjectifs - Comment faire la différenciation et prononciation du masculin et du féminin des adjectifs . **Lettre ou Carte postale** - Comment écrire une lettre ou carte postale en décrivant nouveau logement et nouveau cadre de vie.

Rédaction d'une carte ou d'un message - Comment écrire une rédaction d'une carte ou d'un message de vacances et le climat en France et les cadres de vie (ville et campagne).

UNIT-3 (LECTURE HOURS: 12)

Souvenez-vous

L'imparfait - Comment conjuger les verbes à l'imparfait, les emplois du passé composé et de l'imparfait, expression de la durée, l'enchaînement des idées (alors, donc, mais).

Les moments de la vie - Comment exprimer les moments de la vie, la famille, les relations amicales, amoureuses, familiales.

Raconter un souvenir - Comment raconter brièvement un souvenir, présenter sa famille, faire brièvement la biographie d'une personne.

Demander des informations - Comment demander ou donner des informations sur la biographie d'une personne, sur ses relations amicales ou familiales et interroger quelqu'un sur ses projets.

Spectacles d'un magazine - Comment faire des pages spectacles d'un magazine et une présentation des films sur le thème du couple.

Rédaction de commentaires - Comment écrire des rédactions de commentaires de photos (album souvenirs), le couple et la famille.

UNIT-4 (LECTURE HOURS: 12)

SEMESTER - II
LANGUAGE-THEORY
FRENCH II

On s'appelle ?

Les pronoms compléments - Comment utiliser les pronoms compléments directs et indirects de personne, l'expression de la fréquence et de la répétition.

Les moyens de communication - Comment parler les moyens de communication (courrier, téléphone, internet). **Aborder quelqu'un** - Se présenter, faire valoir son droit, et exprimer une opinion sur la vérité d'un fait. **Messages de voeux** - Comment écrire des messges de voeux, souhaits, remerciements, félicitations, excuses. **Rédactions de petits messages** - Comment écrire des rédactions de petits messages en relation avec ceux qui ont été étudiés en lecture.

Conseils de vivre - Quels sont les conseils de savoir-vivre en France.

UNIT-5 (LECTURE HOURS: 24)

Un bon conseil ! et Parlez-moi de vous !

Expression du déroulement de l'action - Indiquer les actions par les temps passé récent, présent progressif, futur proche, action achevée ou inachevée et les phrases rapportées.

Les Corps - Comment indiquer les corps, la santé et la maladie en français, parler de ses activités de loisirs. **Prendre un rendez-vous** - Comment prendre un rendez-vous par téléphone et exposer ou réagir un problème. **Rythme des constructions** - Quels sont les rythmes des constructions négatives, du discours rapporté, des contractions avec pronoms.

Rédaction de conseils - Comment faire le bref exposé écrit d'un problème personnel, des extraits de magazines, des instructions.

Conseils aux situations d'urgence - Quels sont les conseils pour faire face aux situations d'urgence.

La formation des mots - Comment faire la place de l'adjectif, la proposition relative finale avec "qui", Impératif des verbes avec pronoms.

La description physique - Comment faire la description physique et psychologique des personnes, les vêtements, les couleurs.

Exposer un problème personnel - Comment exposer ou donner des conseils à quelqu'un qui a un problème personnel (santé, relations, etc.,).

Donner une explication - Comment prendre un rendez-vous, demander ou donner une explication. **Différenciation de genre** - Comment faire une différenciation entre masculin ou féminin , descritption de comportements.

Civilisation - Quelques styles comportementaux et vestimentaires en France, se présenter par écrit.

Text Books :

Echo – A1- Méthode de français avec la collaboration de C.GIBBE | Edition:1 | CLE International | J.GIRARDET AND J.PECHEUR (2011)

Reference Books :

R1. Collins dico | Edition:8 | | MAURY-IMPRIMEUR SA MALESHERBES | Robert et Collins(2006)

SEMESTER - II
LANGUAGE-THEORY
HINDI II

OBJECTIVES :

Rashtrabhasha ke prati jagarookh utpann karana. Hindi ki avashyakata aur bolchal hindi ka gyan badana. Hindi ko apna vyavasaya kaise banana, aur baaazaar me iski kya upayogitha hai uska gyan cchatron ko dena.

HOURS / WEEK - 6		
HOURS / SEMESTER - 90		
CIA	**CO-CURRICULAR ACTIVITIES**	**LECTURE**
12	6	72

UNIT 1 (LECTURE HOURS: 18)

Ekanki

Andher nagari - Bharathentu Harishchandra - 1.Ekanki ka udbhav evm vikas, 2.ekanki parichy, 3. patra parichay,.4.andher nagari ka varnan5. guru aur shishy ka nagar gaman6. govardanlal ki lalachi7. andher nagari me andher vichar8.. chupat raja ka rajya9.raja ka galat physala10. raja ke dwara bekasooron ko phansi ki saja 11. goverdan ka phasna12.guru ka agaman13. guru ke dwara shishy ka bachna14. anth me raja shooli par chadna15. chaupat raja ka anth..

Shivaji ka saccha swaroop - Shri.Seth. Govindadas - 1lekhak parichay2. shivaji ki dharana3. mahilavon ki prati sammananiya drishti 4. jeet ka samachar5. ahmed ke putravadhu ko bandhi bana lana6.shivaji ka krodh7. shivaji ke naari ke prati ka soch. maathavon behanon ke roop me dekhne ki roodi8. putravadhu se maufi maangna9. pheshavar ko dand sunana10. dharm nirapekshy rajneeti.

UNIT 2 (LECTURE HOURS: 18)

Ekanki

Do kalakaaar - Shri Bhagavaticharan varma. - 1. Lekhak parichay 2. Do kalakaar. 3.kavi aur chitrakaar ki dayaneeya stithi 4. Paise yintneme chalaki dikhaana 5. Prakhashak aur raies ko sabak sikhana 6. Jeene ka naam jindagi hai 7. udheshya evm nishkaarsh.

Desh ki pukaar- Girish Pankaj - 1. Lekhak parichay.2.raj nethavon ki kudrishti3. kartavya ke prati muh modna3.desh ke prati nahi balki apne baare me jyada sochna. 4.Nishkarsh evm uddheshy

UNIT 3 (LECTURE HOURS: 15)

kavita

1.Nirala-kavi keh gaya hai 2.Nagarjun-tumne kaha tha 3.Mythilicharan Gupt-sakhi ve mujse kehkar jaate 4. Pant-pratibimb 5.Saxena-Macchili - 1.kavi parichay, prakruti par kavi ka drishtikon, jindagi ka soonapan. 2.kavi parichay, nehru ke naa likhi gayi kavita hai 3.kavi parichay, rahul janani yashodhara ka dukhad gatha ka varnan 4.kavi parichay, prakruti me apni maa ka darshan karne ki kala 5.kavi parichay, jeev jantuvon me bhi bhav hai. iski jalak hai is kavita me

UNIT 4 (LECTURE HOURS: 15)

kavita

Dhoomil ki antim kavita 7.Arun Kamal-Vaqt 8.Ramdaras Mishra.-chidiya, cchitti 9.Agnishekar- mere bete ki kavita 10.Jayaprakash Manas-neend se cchotthe hi chala javunga Idhar bahut din hue - .kavi parichay, samanya logon ki vedana ka darshan 7.kavi parichay, manav me manavata ka abhav 8.vaqt badalta rahata hai.. saath hi manav ka drishtikon me bhi parivartan 9.kavi parichay, kashmiriyon ki manodasha 10.kavi parichay, desh ke prati aaj ki stiti par vedana prakat hona..

UNIT 5 (LECTURE HOURS: 6)

Vyakaran

Shabda vichar - Samanaartak shabdh, viparirartak shabdh, kaal badalna vachya badalna... adi...

Anuvaad

Hindi se Angregi - Poore das abyaason ka parichay evm uske anuvad karne me madad karna.

SEMESTER - II
LANGUAGE-THEORY
HINDI II

Text Books :

pushpa kunj | Edition:2 | vibhag dwaara sankhalith granth | BOSMEMBERS(2014)

Reference Books :

Hindi Sahitya ka ithihas | Edition:1 | Lokbharati Prakhashan | . Lakshmisagar Vaarshneya(2001)
Sanskaran | Edition:1 | Manasarover pratam chaar bhag | Premchand(2004)

SEMESTER - II
LANGUAGE-THEORY
MALAYALAM II

OBJECTIVES :

1. To enable the students to learn and imbibe the values, culture and ethics through comprehensive learning of the selected screen play, drama and autobiography.

HOURS / WEEK - 6		
HOURS / SEMESTER - 90		
CIA	CO-CURRICULAR ACTIVITIES	LECTURE
12	6	72

UNIT - 1 (LECTURE HOURS: 16)

Poem

karuna - The Karuna was the last work of Kumaranasan published in 1924. By using limited words vast message is conveyed and as quoted by P.K. Balakrishnan there is no equal work like Karuna which is loaded with tons of messages but inked in very few words. The origin of Karuna is based on the true story - The Gospel of Buddha by Dr.Paul K

UNIT - 2 (LECTURE HOURS: 12)

Poem

Onnam Bhagam - This chapter deals with the introduction of Vasavadutta, a prostitute

Randam Bhagam - The current portion of the poem encompasses the tragedy happened to Vasavadutta and finally she is standing in front of the mouth of death

Moonnam Bhagam - Upagupta, a Buddhist saint giving the light of real meaning of human beings life and after receiving that Vasavadutta accepts death with satisfaction and peace .

UNIT - 3 (LECTURE HOURS: 12)

Travalogue

Himavante Mukalthattil - Artist Rajan Kakkanadan was born in 1942. He worked in Tamil Nadu, Rajasthan, and Bengal & Mumbai. This travelogue is based on the travel done by the author in 1975.He was hero of the famous movie 'Esthappan' directed by Rajan Aravindan.His elder brother Kakkanadan is a doyen Malayalam literature.

UNIT - 4 (LECTURE HOURS: 18)

Travalogue

Travalogue Progession - The experiences of this travelogue are quite unbelievable and fully packed with adventurous moments. The hundreds of miles of this journey with just his shadow as a sole companion encompass fear, pride and dangerous passages. This travelogue consists of memoirs of the preceding journey.

UNIT - 5 (LECTURE HOURS: 14)

Grammar

Upanyasarachana - The art of writing about a concept/topic.

Asayavipulanam - The expansion of old sayings.

Tharjhima - The translation from English to Malayalam

Text Books :

Gadhyashilpam | Edition:2 | Kerala Bhasha Institute | Vasudeva BhattathiriC V(2000)
Himavante Mukalthattil | Edition:1 | Poorna Publications | Rajan Kakkanadan(2013)
Karuna | Edition:3 | Kerala University | KumaranasanN(2002)

Reference Books :

Malayalakavithaa sahithyacharithram | Edition:2 | D C Books | Leelaavathi M(2001)

SEMESTER - II
ENGLISH-THEORY
ENGLISH II- COMMUNICATIVE ENGLISH

OBJECTIVES :

To enable students to develop their listening skill so that they may appreciate its role in the LSRW skills approach to language and improve their pronunciation
To equip students with necessary training in listening so that they can comprehend the speech of people from different backgrounds and regions

HOURS / WEEK - 6		
HOURS / SEMESTER - 90		
CIA	**CO-CURRICULAR ACTIVITIES**	**LECTURE**
12	6	72

UNIT I- THE POWER OF WORDS (LECTURE HOURS: 13)

1. Any Day Words For Everyday Use

Introduction - The beauty of possessing a sound vocabulary gets enhanced if understood in the right perspective. **New words and its meanings** - Ambulatory - Able to move, August - Dignified, Timid - Scared **Meanings and Sentences** - Insolvent - Bankrupt The company has been declared insolvent

2. Divide and Conquer

Introduction - The words you consider long are combinations of short words or parts of words written together

PSEUDOPESUDOHYPOPARATHYROIDISM - Pseudo-Imitate Pseudopesudo-Large imitate Hypo- Lazy Thyriod- Organ in Human Body

PNEUMOENCEPHALOGRAPHY - Pneumo-relating to the lungs Encephalo-relating to the brain Graphy-something written about the particular subject.

3. Words That Make Headlines

Introduction - When you see a new or familiar word in a headline, read on. You are sure to have it explained in the first sentence of the news story.

Meaning and frame a sentence - i)Ambiguous- Doubtful Eg. He seemed ambiguous to take decision in front of the media.

Meaning - ii) Condone-Pardon Eg. We do not condone killing people.

4. Mistaken Identities

Introduction - In English there are many words that are similar in spelling or pronunciation and sometimes we mistakenly reach for a word because it seems to have the right sound or look.

Similar in pronunciation but different in spelling and meaning - Altar-Alter Brake-Break Canvas-Canvass Dual-Duel

5. The Right Word

Introduction - This chapter shows the value of using the right word in the right place

Right word and right meaning - i) Annoy-a little angry Eg. The decisions really annoy him. ii) Carat- Unit of weight for precious stones and pearls iii) Allusion-Illusion iv) Foreword-Forward v) Elicit-Illicit

UNIT II - DEVELOPING READING SKILLS (LECTURE HOURS: 13)

1. A Tale of Seven Berries

Introduction - A shepherd and his herd were responsible for the discovery oh the coffee plant , a native of Koffa district in Abyssinia or Ethiopia.

The Smuggled Seeds - A pilgrim called Baba Budan, also known as Hazarat Shah Janab Allah Magatabi, smuggled seven seeds of coffee in his tunic while he was on a pilgrimage from Mecca .

i)The Local Rulers - i) The local rulers, Mysore Maharaja Krishna Raja Wodeyar III gave away lands and established norms.

ii)19th Century - ii) In the 19th Century, the Maharaja leased the collection of coffee to the British . This was just the beginning of the plantation story as coffee cultivation soon changed hands from the locals to the colonial powers . **Health and Pharmacology** - Scientific studies have examined the relationship between coffee consumption and an array of medical conditions. Findings have been contradictory as to whether coffee has any specific health benefits , and results are similarly conflicting regarding the negative effects of coffee consumption

SEMESTER - II
ENGLISH-THEORY
ENGLISH II- COMMUNICATIVE ENGLISH

Ice-cold-Frezza coffee. - Caffeine is the major coffee constituent which the coffee tolerance or intolerance depends on . In a healthy liver , the majority of caffeine is degraded by the hepatic microsomal enzymatic system . Caffeine is mostly degraded to paraxanthine substances, partially to theobromine and theophylline, and a small amount of unchanged caffeine is excreted by urine

Exercise - Exercises 1 to 50

2. Championing peace

Introduction - For once, this year, let's play the game a little differently. Instead of predicting the future, let's be proactive and see if we can shape it.

Happening of violence - Over the last fifty years, man has became a victim of nuclear weapons, biochemical arms of mass destruction, bloodshed and violence.

The Future is Now - In an era where the internet has grown and has become an indispensable part of our daily lives breaking barriers and creating world citizens, a call for peace is already a happening phenomenon.

Predictions - The future seems bleak and impossible to predict but it is possible to think.

Global peace - Global peace is not one's imagination and all efforts must be energized and sustained by the common man across all nations to champion peace.

Exercises - Exercises 1 to 30

UNIT III - DEVELOPING READING SKILLS (LECTURE HOURS: 10)

1. When words kill

Introduction - Issues dealing with verbal abuse in relationships.

A tool for control - Too many women accept men's excuses for their verbal abuse. They did not realize that some men would always engage in verbal abuse.

Verbal Abuse - One needs a lot of emotional strength to fight against verbal abuse.

The hang up - Until the family and society at large, gets over their hang up, that husbands' verbal abuse is normal and can be tolerated, these men aren't going to reform either.

Communication classes - Men can be helped immensely by anger management classes, couples communication classes or therapy.

2. Human Behaviour Found in Animals

If only animals spoke… - Many observations on the animal behaviour confirm that "animals just lack speech". **Observations on animal behaviour** - A limping individual headed towards an abandoned den . His shoulder bore a deep wound. A big black male enters the den and regurgitating a big chunk of meat to the wounded animal and left . **Observations made on both howler monkeys or chimpanzees** - Females of the group gather around a female who has just given birth to examine the newborn, trying to touch it or take it in their arms. Pretty human. Chimps' behaviour is a mirror of our own.

Animals' greeting behaviour. - Their greeting has the role of establishing the sex and/or the rank of the greeted individual. When two dogs meet, they sniff each other's snout and genitalia

Animal behaviour. - The offspring of the animals play in a way that is very similar to that of humans . Lion cubs play with the tails of the adults, or the dolphin offspring play with shells, which they throw with their fins and reject with their fluke. The play fights in the case of the animal offspring have the role of establishing a hierarchy , which once established can be forever

UNIT IV - DEVELOPING WRITING SKILLS (LECTURE HOURS: 12)

1. Descriptive writing

Sentences and Paragraphs - Descriptive writing is used in essays, reports, fiction, nonfiction and poetry.

Purpose of Descriptive writing - Purpose of Descriptive writing is to make the reader feel, see, taste, smell or hear what you are talking about?

Descriptive Paragraph writing - A descriptive Paragraph is a paragraph, which has sentences that work together to describe and give a clear picture of a person, event, place or thing.

Paragraph writing - Description of objects

2. Narrative writing

Introduction - Writing a Narrative composition appeals to one of human being's instincts , the impulse to share stories. **Elements of Narrative essays** - The focus of a narrative essay is the plot, which is told using enough details to build to a climax.

Time Relationship Transitions - After, afterwards, before, during, earlier, eventually, first, in the meantime, later, meanwhile, next, now, once, second, soon, sooner, then, until, when

Using Connotative Language, Dialogue, Beginnings and Endings - The language used in narrative writing helps the reader imagine himself in the events you tell about. When writing a narrative paragraph about one event in your life, the use of dialogue will definitely make the event more vivid to the readers

SEMESTER - II
ENGLISH-THEORY
ENGLISH II- COMMUNICATIVE ENGLISH

3. Note making

Introduction - Note making should essentially, constitute brevity, no proper sentences need be made but what is to be answered is that, the note should contain enough information to remind one of the whole passage when these notes are looked at later on.

Structure of Notes - 1. The relevance of note-making 2. Steps in note-making 3. Features of notes 4. Intellectual activity **Skimming and Scanning** - 1. The Main theme and 2. The Main topics

1.Abbreviations and Symbols 2.Condensing Information - 1.For long words and words repeatedly used in a passage, abbreviations should be used. 2. Use of abbreviation and symbols, Using tables and diagrams, etc.

UNIT V - DEVELOPING WRITING SKILLS (LECTURE HOURS: 12)

1. Creative writing

Introduction of Slogan writing - A slogan is a catchy word or a short memorable advertising phrase which creates an awareness of the subject of the slogan and incites the reader to respond accordingly.

Slogans - Slogans are used to 1. Create Awareness 2. To evoke response 3. To highlight a merit

Caption Writing - Captions are single sentence headings or titles accompanying a picture or a photograph.

i) News Captions ii) Creative Captions - i) News captions are those which accompany a photograph or an event in a newspaper/magazine. ii)Creative captions which aim at inciting response to a mood or emotion depend on style and expression of the language.

2. Comprehensive Exercises

Read the passage - Read the passage fairly quickly to get the general idea.

Given passage - Read again, a little slowly, so as to know the detail.

Same Passage - Study the questions thoroughly, turn to the relevant portions of the passage, read them again, and then rewrite that in your own words, neatly and precisely

Exercises - Use complete sentences

3. Precis Writing

Introduction - The nature of Precis-writing is indicated by its name, which is derived from the French word 'precise', which means 'an abstract'.

Precis - 1. Read through the passage 2.Read it again carefully 3.Read it a third time and 4. Connect the arguments into paragraphs

Notes on Precis work - i) Using your own words ii) Re-reading and Correct Linking iii) Keeping a Balance, counting the words

Alternative to Precis - It is important to remember that these alternative ways are methods employed to test your summarizing skills.

UNIT VI (LECTURE HOURS: 2)

Tutorial-1
Tutorial - Debate
Tutorial-2
Tutorial - Quiz

UNIT VII (LECTURE HOURS: 2)

Tutorial-1
Tutorial - Debate
Tutorial-2
Tutorial - Quiz

UNIT VIII (LECTURE HOURS: 2)

Tutorial-1
Tutorial - Debate
Tutorial-2
Tutorial - Quiz

UNIT IX (LECTURE HOURS: 3)

Tutorial-1
Tutorial - Quiz
Tutorial-2
Tutorial - Debate
Tutorial-3
Tutorial - Quiz

SEMESTER - II
ENGLISH-THEORY
ENGLISH II- COMMUNICATIVE ENGLISH

UNIT X (LECTURE HOURS: 3)

Tutorial-1
Tutorial - Quiz
Tutorial-2
Tutorial - Debate
Tutorial-3
Tutorial - Quiz

Text Books :

Developing Reading Skills | Edition:1 | Bloomsbury | Dr. HyacinthPink(2015)
Developing Writing Skills | Edition:1 | Bloomsbury | Dr.HyacinthPink(2015)
The Power of Words | Edition:1 | Bloomsbury | Dr. HyacinthPink(2015)

Reference Books :

Oxford Advanced Learner's Dictionary | Edition:1 | Oxford University Press | A SHornby(2010)

SEMESTER - II
MAJOR-THEORY
OBJECT ORIENTED PROGRAMMING USING JAVA

OBJECTIVES :

To inculcate knowledge on Java Programming concepts .

To understand the basic concepts of class, objects, constructors , Inheritance ,Polymorphism , Control Structures ,Arrays, String ,Date, Debug and Exception handling

· Apply the techniques of object oriented structured pattern and packages to break a program into smaller module and describe the process of package, Interface and Files.

Develop Desktop applications in java using NetBeans IDE.

HOURS / WEEK - 5		
HOURS / SEMESTER - 75		
CIA	**CO-CURRICULAR ACTIVITIES**	**LECTURE**
10	**5**	**60**

UNIT-I (LECTURE HOURS: 13)

INTRODUCTION TO JAVA

Tool Kits and Platforms - Java Development Kit(JDK) Software Development Kit(SDK)

How Java compare with C++,c# - Java TimeLine Operating Systems Support by Java Compare to C++ and C# (Syntax, Platforms, Speed and Memory) Description - SDK and JDK version

Applications, Applet and Servlets - Graphical User Interface(GUI)Application and Console Application Special Type of Web Based Application - Applet Enterprise Databases Application are based on servlet

How Java Compiles and interprets code - Java IDE- Source file Java Compiler , Byte Code Java Virtual Machine (JVM-Interpreter) -OS

Introduction to Java IDEs - Popular Java IDE's Features Provided by the IDE's

HOW TO USE NETBEANS TO DEVELOP NEW PROJECT

How to create a new project - New Java Application Dialogue Box Name and Location for the Project **How to set java version for a project** - The Project Properties Dialogue Box

How to work java source code and files - Code Editor with the Starting Source code for the Project Save Source code, Rename and Delete

How to use the Code Completion feature - Code Compilation Features Prevent typing Mistakes Methods and fields are available **How to Detect and correct syntax errors** - Detect Syntax errors Display red error icon

HOW TO USE NETBEANS TO WORK WITH EXISTING PROJECT

How to open close and delete a project - Open project dialogue box Locate and select the project or Open recent project command Delete source file

How to compile and run a project - Run the project button in the tool bar or F6

How to Use the Output window with a console Application - To run an Application that request input from the console Display Message the application is Compiled Display errors that are encountered

How to Work with two or more projects - Open and work two or more project at a time Right click on the file and run the file command

BASIC CODING SKILLS

How to code statements - The statements specify the sequence of actions to be performed Coding format Coding standard (How to name a class method and variables)

How to Code comments - Single line comments Multiple line comments or Block comments Importance of comments line **How to Create Identifiers** - Valid Identifier Rules for naming an identifier Keywords

How to declare a class and a main method - Syntax for declaring a class, main method Rules for naming a class and method

HOW TO WORK WITH NUMERIC VARIABLES AND STRING VARIABLES

How to Declare and Initialize Variables - Primitive data types Syntax to create and initialize variables Code Assignment statements Naming recommendations for variables

Code assignment statements - Simple assignment statement

Code arithmetic expressions - Basic arithmetic Expression Statements that use simple arithmetic expression Increment Counter variable Mix integer and double variable Type casting

How to Create a String Object - Introductions about String class String literal

Join and append a Strings - Syntax for declaring and initializing String variable How to join a string Join string and a Number How to append one string to another string with the +Operator, += operator

SEMESTER - II
MAJOR-THEORY
OBJECT ORIENTED PROGRAMMING USING JAVA

Include Special characters in Strings - Common Escape Sequences Newline, Tab and returns Quotation mark and back slash

HOW TO USE
JAVA CLASSES, OBJECTS AND METHODS

How to Create Objects and call methods - Create an Object from a class Call a method from an object Call Static method from class

Import java classes - Java API Packages Syntax of the import Statement Use the scanner class to create Object **API**

Documentation - http://docs.oracle.com/javase/7/docs/api/

HOW TO USE THE CONSOLE INPUT AND OUTPUT

How to Use of System. out object to print output to the console - Two methods of System.out Object, The println method , The print method, Example: An application that print data to the console

Use of Scanner class to read input from the console - Import util package The scanner class create a scanner object common methods of a scanner object use the methods of scanner object

Example that get input from the console - Example 1: Create a class Invoice App Gets three values from user using the scanner object The values are productcode, price, quantity Calculate and display the result in total variable Example 2: create a class TestScoreApp Read three values from one line using scanner object Calculate the average and display the result

HOW TO CODE SIMPLE CONTROL STATEMENTS

How to Compare numeric variables - Boolean expression Relational operators to compare integer and double data type **Compare String Variables** - Relational operators Conditional expressions Two methods of string class

If /else Statements - The syntax of the IF/else statement IF statement without ELSE IF statement with ELSE class IF statement with ELSEIF and ELSE class

While Statements - Syntax of the while loop A loop continues while choice is "YES" A loop that calculate the sum of numbers 1 to 4

BASIC SKILLS FOR WORKING WITH DATA

Eight Primitive Data Types - Primitive data types and their Range

Declare and Initialize variables - Syntax -Example Declare and initialize a variable in two statements Declare and initialize variable in one statement Naming conversion

Declare and Initialize constants - Declare and initialize a constant Naming conversion

How to code Assignment Statements and Arithmetic Expressions - Arithmetic operators Example of simple assignment statement **How to use Shortcut Assignment Operators** - Shortcut assignment operators

The Order of Precedence - Order Precedence of arithmetic operators Example programs

Work with Casting - How to implicit casting work How to code an explicit cast Cast between char and integer types

HOW TO USE JAVA CLASSES FOR WORKING WITH DATA TYPES

How to use the Number Format Class - Number format class Three static methods of the number format class Three methods of a number format class

Use the Math Class - Math class Common static methods of the math class Example : Use math class to perform round, pow, sqrt, max ,min and random methods

Integer and Double Classes - Constructor integer and double, Static methods of integer and double class , How to create integer and double object , Convert primitive type to string objects, String object to primitive types

UNIT-II (LECTURE HOURS: 11)

BOOLEAN EXPRESSION

Compare primitive data types - Relational operators Example of Boolean expression

Compare String - Two methods of string class Expressions that compare to string values

Logical Operators - Logical operators Example program

IF/ELSE AND SWITCH STATEMENTS

If/Else Statements - IF/ELSE syntax IF statement with elseif and else clauses If statement that contains two blocks of code Nested if statement

Switch Statements - The syntax of switch statement Switch statement that uses an integer Switch statement that uses a String Switch statement that false throw case labels Example : Import text and util package Create class Invoice Get three input from the user Customertype, subtotal, discount percent, discount amount Use nested if statement To Calculate discount percent Display the result

LOOPS

While and Do - While Loops - How to code a do-while loop Syntax of the while and do-while loop Use while and do-while calculate feature values

For Loops - Syntax of the for loop Example: A for loop that store 0 to 4 in string A for loop that adds the number 8 6 4 2 A for loop that calculate the future value

Nested Loops - Nested loop that print a table of future value

BREAK AND CONTINUE STATEMENTS

SEMESTER - II
MAJOR-THEORY
OBJECT ORIENTED PROGRAMMING USING JAVA

Break Statements - The syntax of the break statement Syntax of the labeled break statement **Continue Statements** - The Syntax of the continue statement Syntax of the labeled continue statement

HOW TO CODE AND CALL STATIC METHODS

Static Methods - Code a static method Access modifier Return type

Call Static Methods - Syntax for coding statics methods Static method with no parameter and return type Static method with three parameter that returns a double value Syntax for calling static method in the same class

Case Study

Create an application that prints data to the console - Future values application with static method

HOW TO HANDLE EXCEPTION

How Exceptions Work - Some of the classes in the exception hierarchy the console after an input mismatch exception Four methods that my throw exception

How to Catch Exceptions - The syntax for try statement Import input mismatch exception class

The Future Value Application with Exception handling - Import util and text package Create a class Future ValuesExceptionApp Use scanner class get the choice from the user Use try catch block throw input mismatch exception

VALIDATE DATA

How to prevent exceptions from being thrown - Methods of the scanner class use to validate data Code that prevents and input mismatch exception

Validate a Single Entry - Data validation Invalid data Display the error message To common type numeric entry **Generic methods to validate an entry** - Methods that gets a valid numeric format A method to check valid numeric range

BASIC SKILLS FOR TESTING AND DEBUGGING

Typical test phases - First phase : user interface Test the application with valid input Test the application with invalid data

Three Types of Errors - Syntax error or Compile -time error Runtime error Logic error The goal of testing goal of debugging

Common Java Errors - Common Syntax errors Problem with identifier problem with values Problem with floating point arithmetic

A Simple way to trace code execution - Println statement to trace execution

USE NETBEANS TO DEBUG AN APPLICATION

How to set and Remove Breakpoints - Code editor window with a break point Remove the break point

Step through code - Step over button Spilt the step over button Continue normal execution Finished debugger session

How to inspect variables - Set break point and step through code the variable window Some of the buttons on the debug toolbar **Inspect Stack Trace** - Debugging session with call stack window Window - debugging - call stack command

UNIT-III (LECTURE HOURS: 13)

AN INTRODUCTION TO CLASSES

class can be used to structure an application - The Architecture of a three tiered application Presentation layer Middle layer Database layer

How encapsulation works - A class diagram for the product class Access fields and methods

The relationship between a class and its objects - How an object created from the a class Create Instance of class Demonstrate relationship between class and its object using product class

CODE A CLASS THAT DEFINE AN OBJECT

Use NetBeans to create a new class - The Dialog box for creating new java class From a package

Code for product class - Create a product class Instance variable Define constructor Use get and set method for getting a values **Instance variables** - The syntax for declaring instance variables Where we can declare instance variable

Constructors - Syntax for coding constructor Constructor that assign default values, Custom constructor with parameter **Methods** - Syntax for coding a method, The method does not accept parameters are return data Get method that Returns values ,Custom get method set method

Overload methods - Method that excepts one argument, An overloaded method that provides a default value, An overloaded method with two argument

Use of this keyword - The syntax for using this keyword , How to refer instance variable, How to refer methods, How to call a constructor, How to send a current object method How to send a current object static method

Use NetBeans to work with classes - NetBeans window for product application that dialogue box generating get and set methods

CREATE AND USE AN OBJECT

Create an Object - How to create an object using one and two line statement

Call the methods of an Object - Syntax A method call with argument and return values

Primitive types and reference types are passes to a method - The primitive type that passed method An object type that passed a methodCode that calls the method

ProductDB Class Case study - Example : Import a util package Create a class product, productDb class, product App Display welcome message Use the scanner class Get one or two products from the user Create product objectDisplay the output

CODE AND USE STATIC FIELDS AND METHODS

SEMESTER - II
MAJOR-THEORY
OBJECT ORIENTED PROGRAMMING USING JAVA

Static fields and methods - How to declare a static fields Class with contains a static constant and a static method A class with static variable and static method

Call Static fields and methods - Syntax for calling static field or method Statement to call static field and static methods **Static Initialization Block** - Static initialization block

Use of static fields and methods - Syntax for coding a static initialization block

AN INTRODUCTION TO INHERITANCE

How inheritance work - Super class Sub class

Java API Uses Inheritance - Inheritance hierarchy for swing form and controls

Object Classes - Object classMethods of the object class

How to use inheritance in your Applications - Product super class Book, Software inherited class

BASIC SKILLS TO WORK WITH INHERITANCE

Create a Super Class - Access modifier Private ,public, protected An annotation for overriding a method Example

Create Sub Class - The syntax creating sub class To declare a subclass To call a super class constructor To call a super class method **How Polymorphism works The Product , Book, Software and ProductDB class** - Three versions of the toString method Code that uses the overridden methods Example: Step 1:Import Scanner Class using Util package Step 2: import text package Step 3: Create Product class Step4 : override the toString method and display code, description and price details Step 5: Create Book and Software classes inherit from Product class Step 6: Override the toString methods and display author and version details Step 7 : Create ProductDB class Step 8: Create a object for Book and software class Step 9 : Create a reference for Product class Step 10: Set the product variable to the Book and Software Object.

SKILLS FOR WORKING WITH INHERITANCE

Get Information about Object's Type - Syntax for class Code that displays and object's typeCode that test object's type **Cast Objects** - Casting examples that use product and book classes with description

Compare Objects - How to equal's method of object class work with example How to override the equal's method that with example

ABSTRACT AND FINAL KEYWORDS

Abstract Keyword - An abstract product class A class that inherits the abstract product class with description **Final Keyword** - A final class, a final method , final parameter with description

AN INTRODUCTION TO INTERFACES

A Simple Interface - A Printable interface that defines a print method. A product class that implements the printable interface Code that uses the print method of the product class

Interfaces compared to abstract classes - Difference between abstract classes and interface A printable interfaceA printable abstract class advantages of an abstract class and an interface

Some Interfaces of Java API - interface in java .lang packageinterface in java.util and java.awt.event package

WORK WITH AN INTERFACE

Code an Interface - Syntax for declaring an interface An interface that defines three methods constants and Tagging interface with no members

Implement an Interface - Syntax for implementing an interface A class employee that implements two interfaces. printable and department

How to Inherit a class and implement an Interface - Syntax for inheriting a class and implementing an interface. A class Book inherit another class product and implement printable interface

How to use an interface as a parameter - A method that accept a printable object Code that passes product object to the method **Use of Inheritance with Interfaces** - Syntax for declaring an interface that inherits other interfaces

Use NetBeans to work with Interfaces - A Class implements an interface productDAO interface that inherits other interfaces

UNIT-IV (LECTURE HOURS: 13)

WORK WITH PACKAGES

An Introduction to Packages - Directories and files for an application that uses packages

Use NetBeans to work with Packages and Libraries - Project contains multiple packages Packages store one or more classes interfaces. A project that uses LibraryHow to create and use Library

USE JAVADOC TO DOCUMENT A PACKAGE

Add Javadoc comments to a class - Javadoc class comments to a class Single and multiple line comments

How to use HTML and javadoc tags in javadoc comments - Common javadoc tags. Create a class product with comments that use HTMLand Javadoc tags

Use NetBeans to generate documentation - Projects window and select the "Generate javadoc" command **View the documentation for a Package** - Sharing web browser and navigate index .html file

WORK WITH ENUMERATIONS

SEMESTER - II
MAJOR-THEORY
OBJECT ORIENTED PROGRAMMING USING JAVA

Declare An Enumeration - Enumeration - set of related constants. Syntax for declaring an enumeration. Create an enumeration that defines three shipping types

Use of an Enumeration - A statement that use the enumeration and one of its constant. A method that uses enumeration as parameter type. A statement that calls the method

Enhance an Enumeration - By using name() ,ordinal() method enhance enumeration Add a method to an enumeration that overrides toString method

Work with static imports - How to code a static import statement

BASIC SKILLS FOR WORKING WITH ARRAYS

Create An Array - Syntax for declaring and instantiating an array Examples for array declarations

Assign values to the Elements of an Array - Syntax for referring to an element of an array Examples that assign values by accessing each element. Syntax for creating an array and assigning values in one statement with example

How to use enhanced for loops with Arrays - Syntax of enhanced for loop. Code that prints an array of prices to the console. Code that computes the average of array of prices

MORE SKILLS FOR WORKING WITH ARRAYS

Methods for the Arrays class - Static methods of the Array class

Code Examples with the Arrays Class - Code that uses fill method Code that uses fillmethod to fill3 elements in an array Code that uses the equals method Code that uses SORT method Code that uses sort and binary search methods **How to create reference and copy to an array** - How to create a reference and how to copy an array

TWO DIMENSIONAL ARRAYS

Working with Rectangular Arrays - Syntax for creating a rectangular array. Assign values for rectangular array Use nested for loops to process a rectangular array

Jagged Arrays - Syntax for creating jagged array. Code that creates a jagged array of integers Code that creates and initialize a jagged array of strings Code that creates and initialize a jagged array of integers Code that prints the contents of jagged array of integers. Code that uses foreach loops to print a jagged array

WORK WITH DATE AND TIME

Use the Gregorian Calendar Class to set dates and times - Gregorian Calendar class Common constructors A statement that gets current date Statements that create dates with literals Statements that create a date with variables

Use of Calendar and Gregorian Calendar fields and methods - The Calendar class Common fields of calendar class Common methods of calendar and Gregorian Calendar classes Code that changes a Gregorian object Code that accesses fields in Gregorian Calendar object

Use of Date Class - Common constructors and methods A statement that converts a Gregorian Calendar object to a Date object A statement that gets a date object for current date/time. Statements that convert Date objects to string and long variables. Code that calculates the number of days between two dates

DateFormat class to format Dates and Times - Common static methods fields and methods, Code that formats a Date objectCode that formats a Gregorian Calendar objectCode that overrides the default date and time formats

STRING CLASS

Create Strings - Common constructors of string class Two ways to create an empty string, a string from another string, a string from an array of characters, a string from an array of bytes

Methods of the String Class - Methods for manipulating stringsMethods for comparing strings

StringBuilder Class

Constructors and methods of the String Builder Class - Constructor of StringBuilder class Methods of StringBuilder class

UNIT-V (LECTURE HOURS: 10)

INTRODUCTION TO EXCEPTIONS

The Exception Hierarchy - Description about exception The throwable hierarchy Common checked exceptions Common unchecked exceptions

How exceptions are propagated - Two ways to handle checked exceptions- throw and catch the exception.

WORK WITH EXCEPTIONS

Use the Try statement - Syntax of try statement A method that catches two types of exception and usage of finally clause . **Try -with- resources statement** - Syntax of try-with-resource statement A method that catches two types of exception and automatically closes specified resource.

Methods of an Exception - Four common methods. Generate a code to print exception data to the error output stream Generate a code to print exception data to the standard output stream

Use of multi-catch Block - Syntax of multi-catch block A method that does not use a multi-catch block A method that use a multi-catch block

SEMESTER - II
MAJOR-THEORY
OBJECT ORIENTED PROGRAMMING USING JAVA

Throws clause - Syntax for declaration of a method that throws exception. A method that throws an IOException with example A method that Catches the exception with example.

Throw statement - Syntax of throw statements Constructors of throwable class A method that throws an unchecked exception Code that thows an IOException for testing purposes. Code that rethrows an exception

INTRODUCTION TO DIRECTORIES AND FILES -INTRODUCTION TO FILE INPUT AND OUTPUT

A packages for working with directories and files - A package for working with directives and files, A static method of the Paths class, Methods of the Path interface, Static methods of the Files class

Code examples that with directories and files - Code that creates a directory if it doesn't already exist, Code that creates a file if it doesn't already exist, code that displays information about a file

How files and streams work - A text file that's opened by a text editor, A binary file that's opened by a text editor, Two types of files, Two types of streams

A file I/O example - Import all necessary packages, Get a Path object for the file, Write data to the file, read data to the file **How to work with I/O exceptions** - A subset of the IOException hierarchy, Common I/O exceptions, code that handles I/O exceptions

HOW TO WORK WITH TEXT FILES

How to connect a character output stream to a file - A subset of the Writer hierarchy, Classes used to connect a character output stream to a file, Constructors of these classes

How to write to a text file - Common methods of the PrintWriter class, Code that appends a string and an object to a text file, Code that writes a Product object to a delimited text file

How to connect a character input stream to a file - A subset of the Reader hierarchy, Classes used to connect to a file with a buffer, Constructors of theses classes, How to connect a character input stream to a file

How to read from a text file - Common methods of the BufferedReader class, Code that reads the records in a text file, Code that reads a Product object from a delimited text file

An interface for working with file I/O - The ProductDAO interface, The ProductReader interface, The ProductWriter interface, The ProductConstants interface

A class that works with a text file - The code for the ProductTextFile class

HOW TO WORK WITH BINARY FILES-HOW TO WORK WITH RANDOM-ACCESS FILES

How to connect a binary output stream to a file - A subset of the OutputStream hierarchy, Classes used to connect to a Output Stream to a file, Constructors of theses classes, How to connect to a file with a Buffer

How to write to a binary file - Common methods of the DataOutput inferface, Methods of the DataOutputStream class, Code that writes data to a binary file

How to connect a binary input stream to a file - A subset of the InputStream hierarchy, Classes used to connect to a binary input Stream to a file, Constructors of theses classes, How to connect a binary input stream to a file

How to read from a binary file - Common methods of the DataInput inferface, Methods of the DataInputStream class, Code that reads Product objects from a binary file

Two ways to work with binary strings - Two ways to write a binary string, Two ways to read a binary string **How to connect to a random-access file** - Constructors of the RandomAccessFile class, Access mode values

How to read to and write from a random-access file - Two interfaces implemented by the RandomAccessFile class, Methods of the RandomAccessFile class used for input and output

How to read and write fixed-length strings - A class that writes and reads fixed-length strings **A class that works with a random-access file** - The code for the ProductRandomFile class

Text Books :

Murach's Java Programming | Edition:4 | Shroff publishers and Distributors Pvt,Ltd. | Joel Murach (2012)

Reference Books :

Head First Java | Edition:1 | O'Reilly Media, Inc | Bert Bates AND Kathy Sierra(2003)

SEMESTER - II
MAJOR-THEORY
OBJECT ORIENTED DESIGN USING UML

OBJECTIVES :

- Enable a requirements model using UML class notations and use -cases based on statements of user requirements.
- Create the OO design of a system from the requirements model in terms of a high-level architecture description.
- Comprehend the nature of design patterns, and to apply these patterns in creating an OO design.
- Learn to transition from analysis to design by using UML.

HOURS / WEEK - 3		
HOURS / SEMESTER - 45		
CIA	**CO-CURRICULAR ACTIVITIES**	**LECTURE**
6	**3**	**36**

UNIT I (LECTURE HOURS: 7)

Introduction

What is the UML? - Definition, Modeling Language, Process ,Explanation

Notations & Meta Models - Concept, UML Meta Model Extract Diagram, Explanation

Why do Analysis and Design? - Communication, Explanation,Learning OO, Advantages. Communicating with Domain Experts, Explanation

An Outline Development Process

Overview of the Process - Basics of UML, High Level View of Development Process, Explanation **Inception** - Concept, Explanation

Elaboration - Risk Categories, Dealing with Requirement Risks, Dealing with Technological Risks, Dealing with Skill Risks, Dealing with Political Risks

Planning the Construction Phase - Steps to Plan an Iterative Project, Process, Release plan , Explanation **Construction** - Purpose of Construction Phase, Refactoring, When the plan goes awry, Using the UML in construction, Explanation

Transition - Concept, Example, Explanation

When to Use Iterative Development - When to Use Iterative Development, Explanation

Case Study

Case Study - Diagrammatic Representation of a Dice Game Using Conceptual Model

UNIT 2 (LECTURE HOURS: 8)

USE CASES

Use Cases - Basics of Use Cases, Scenario, Use Case Text Example

Use Case Diagram - Concept, Example, Actor, Relationships, Diagram

When to Use Use Cases - When to use Use Cases, Explanation

Class Diagram: The Essentials

Class Diagram - Definition, Kinds of Static relationship, Diagram

Perspectives - Conceptual, Specification, Implementation, Explanation

Associations - Definition, Association End, Role Name, Multiplicity, Navigability with Diagram, Unidirectional and Bidirectional association, Explanation

Attributes - Definition, Syntax, Example, Explanation

Operations - Definition, Syntax, Example, Getting Method, Setting Method, Difference between Operation and Method, Explanation

Generalization and Constraint Rules - Concept, Example, Explanation, Basic Constructs, Rules

When to Use Class Diagrams - When to Use Class Diagrams, Tips, Explanation

Interaction diagrams

Interaction Diagrams - Definition, Concept, Behavior

Sequence Diagrams - Concurrent Process and Activations, Asynchronous Messages, Deletion with Diagram

Collaboration Diagram - Concept, Diagram with Simple and Decimal Numbering

Comparing Sequence and Collaboration Diagrams - Principle Features, Problems of Interaction Diagram, CRC Cards **When to Use Interaction Diagram** - When to Use Interaction Diagram, Explanation

Case Study

SEMESTER - II
MAJOR-THEORY
OBJECT ORIENTED DESIGN USING UML

Case Study - Use Case diagram & Class Diagram for ATM, Sequence Diagram & Collaboration Diagram for ATM

UNIT 3 (LECTURE HOURS: 6)

Class Diagrams: Advanced Concepts

Stereotypes - Definition, Interface, Profile

Object Diagram and Class Scope Operations & Attributes - Definition, Diagram, Class Scope Notation **Multiple Classification** - Classification, Single, Multiple Classification, Discriminator, Diagram **Dynamic Classification** - Dynamic, Static Classification, Diagram **Aggregation and Composition** - Definition, Diagram with Explanation

Derived Associations and Attributes - Concept, Diagram, Time Period Class, Explanation

Interfaces and Abstract Classes - Basic Concept, Example, Realization, Dependency, Notation for Interface **Reference Objects and Value Objects** - Definition, Example, Difference between Reference and Value Objects **Collection for multivalued Association Ends** - Definition, Example, Explanation

Frozen - Meaning, Example, Distinction between Read Only and Frozen

Case Study

Case Study - Diagrammatic Representation for Hospital Management Using Multiple & Dynamic Classification, Aggregation and Composition

UNIT 4 (LECTURE HOURS: 7)

Packages and Collaborations

Packages and Collaborations - Basic Concept, Functional Decomposition, Packages **Packages** - Package Diagram, Dependency, Diagram, Advanced Package Diagram, Explanation

Collaborations - Definition, Sequence Diagram for Making a Sale, Explanation, Collaboration in a Package, Class Diagram for Make Sale Collaboration

When to Use Package Diagrams and Collaboration - When to Use Package and Collaboration, Explanation

State Diagrams

State Diagrams - Definition, Diagram, Actions, Activities, State Diagram Without Super State, Explanation, State Diagram With Super State, Explanation

Concurrent State Diagrams - Concept, Payment Authorization Diagram , Explanation **When to Use State Diagrams** - When to Use State Diagram, Explanation

Case Study

Case Study - Diagrammatic Representation for Order Management Using Packages || Diagrammatic Representation for Order Management Using State diagrams

UNIT 5 (LECTURE HOURS: 8)

Activity Diagrams

Activity Diagrams - Basic Concept, Activity State, Diagram ,Conditional Behavior, Branch, Merge, Fork, Conditional Thread, Diagram

Decomposing an Activity - Concept, Diagram: Composite Activity for Delivery, Explanation

Dynamic Concurrency - Definition, Example,

Swimlanes - Definition, Diagram, Explanation

When to Use Activity Diagram - Strength and Weakness of Activity Diagram , Use and Don't Use for following Situations, Explanation

Physical Diagrams

Deployment Diagrams - Definition, Node, Connection, Diagram **Component Diagrams** - Definition, Dependency, Diagram, Explanation

Combining Component and Deployment Diagrams - Combine Component and Deployment, Explanation **When to Use Physical Diagram** - When to Use Physical Diagram, Explanation

Case Study

Case Study - Diagrammatic Representation for Hall Ticket Generation Using Activity Diagram

Text Books :

UML Distilled | Edition:Second | Pearson Education | Kendall Scott AND Martin Fowler (2003)

Reference Books :

Object - Oriented Modeling and Design with UML | Edition:2 | Pearson Education | James R Rumbaugh AND Michael R Blaha ( )
Object Oriented Analysis and Design with applications | Edition:2 | Pearson Education | Grady Booch(2000)

SEMESTER - II
MAJOR-PRACTICAL
PROGRAMMING LAB -II (JAVA PROGRAMMING)

OBJECTIVES :

The goal of this paper is to provide students with the ability to write programs in Java and apply concepts described in the Object-Oriented Programming. To understand the basic concepts of problem solving approaches and to develop programs with java coding standards. Apply the techniques of basic decision making control structures, functions, class, objects, OOP features, Array, String , Interface .

HOURS / WEEK - 5		
HOURS / SEMESTER - 75		
CIA	**CO-CURRICULAR ACTIVITIES**	**LECTURE**
10	**5**	**60**

I (LECTURE HOURS: 12)

CALCULATE A RECTANGLE'S AREA AND PERIMETER

The application prompts the user to enter values for the length and width of a rectangle - Aim **The application displays the area and perimeter of the rectangle** - Algorithm **The application prompts the user to continue.** - Code

II (LECTURE HOURS: 12)

CALCULATE INTEREST

The application prompts the user to enter a loan amount and an interest rate - Aim
The application calculates the interest amount and formats the loan amount, interest rate, and interest amount. Then, it displays the formatted results to the user - Algorithm
The application prompts the user to continue - Code

III (LECTURE HOURS: 12)

GUESS A NUMBER FROM 1 TO 100

The application prompts the user to enter an int value from 1 to 100 until the user guesses the random number that the application has generated - Aim
The application displays messages that indicate whether the user's guess is too high or too low - Algorithm
When the user guesses the number, the application displays the number of guesses along with a rating. Then, the application asks if the user wants to play again When the user exits the game, the application displays a goodbye message. - Code

IV (LECTURE HOURS: 12)

CALCULATE A CIRCLE'S CIRCUMFERENCE AND AREA

The application prompts the user to continue
When the user chooses not to continue, the application displays a goodbye message that indicates the number of Circle objects that were created by the application
The application prompts the user to enter the radius of a circle - Aim
If the user enters invalid data, the application displays an appropriate error message and prompts the user again until the user enters valid data - Algorithm
When the user enters a valid radius, the application calculates and displays the circumference and area of the circle to the nearest 2 decimal places - Code

V (LECTURE HOURS: 12)

CALCULATE A PLAYER'S BATTING STATISTICS

After all the at-bat results are entered, the application displays the batting average and slugging percent
This application calculates the batting average and slugging percentage for one or more baseball or softball players - Aim **For each player, the application first asks for the number of at bats. Then, for each at bat, the application asks for the result** - Algortihm
To enter an at-bat result, the user enters the number of bases earned by the batter. If the batter was out, the user enters 0. Otherwise, the user enters 1 for a single, 2 for a double, 3 for a triple, or 4 for a home run - Code

SEMESTER - II
MAJOR-PRACTICAL
PROGRAMMING LAB -II (JAVA PROGRAMMING)

Text Books :

Murach's Java Programming | Edition:4 | Shroff publishers and Distributors Pvt,Ltd | Joel Murach(2012)

Reference Books :

Head First Java | Edition:1 | O'Reilly Media, Inc USA | Bert Bates AND Kathy Sierra(2003)

SEMESTER - II
ALLIED-THEORY
MATHEMATICAL FOUNDATION FOR TECHNOLOGY

OBJECTIVES :

On successful completion of this course the students should gain knowledge about the mathematical logic , relations and graphs

HOURS / WEEK - 5		
HOURS / SEMESTER - 75		
CIA	**CO-CURRICULAR ACTIVITIES**	**LECTURE**
10	**5**	**60**

UNIT I (LECTURE HOURS: 12)

MATRICES

Introduction - Basic definitions
Determination - Related Problems
Inverse of a matrix - Related Problems
Rank of a Matrix - Related Problems
Eigen value Problems - Related Problems

UNIT II (LECTURE HOURS: 12)

MATHEMATICAL LOGIC

Introduction - Basic definitions
Prepositional calculus - Variables & constants

BASIC LOGICAL OPERATION

Types of logical operations - Truth tables
Conditional & Biconditional statements - Truth tables
Tautologies & Contradiction - Related Problems

ARGUMENT

Fundamental rules - Valid or invalid problems
Methods of proof - Examples of direct & indirect problems

UNIT III (LECTURE HOURS: 12)

SET THEORY

Introduction - Basic definitions
Set and its elements - Standard sets & symbols
Set description - Roster & sets builder forms
Types of Sets - Definitions of various sets
Venn-Euler Diagrams - Diagrams of various sets
Set operations and Laws of Set theory - Related Problems

FUNDAMENTAL PRODUCTS

Partitions of sets - Basic definitions
Algebra of sets and Duality - Related Problems
Inclusion and Exclusion Principle - Related Problems

UNIT IV (LECTURE HOURS: 12)

RELATIONS

Cartesian product of sets - Related Problems
Binary relations - Definition & related problems
Set operations on relation - properties
Partial order relation - Related Problems
Equivalence relation - Related Problems
Composition of relation - Matrix representation

SEMESTER - II
ALLIED-THEORY
MATHEMATICAL FOUNDATION FOR TECHNOLOGY

FUNCTION

Introduction - Definition
Types of functions - various types
Composition of functions - Definition & related problems

UNIT V (LECTURE HOURS: 12)

GRAPH THEORY

Basic Terminology - Definition
Paths-Cycle & Connectivity - Types & Examples
Sub graphs - Diagram representation
Types of graphs - Definition
Representation of graphs in computer memory - Matrix representation

TREES

Properties of trees - properties
Binary trees - Types & Examples
Traversing Binary trees - Types & Examples
Computer Representation of general trees - Types & Examples

Text Books :

T1.Engineering Mathematics | Edition:5 Edition | J.J.Publication | M.K.VENKATARAMAN(1999)
T2.Discrete Mathematics | Edition:2 Edition | Macmillan India Ltd | J.K.SHARMA(2007)

Reference Books :

R1.Discrete Mathematics | Edition:1 Edition | The National Publishing Company | DR.M.K.VENKATARAMAN,N.SRIDHARAN & N.CHANDRASEKARAN(2009)

SEMESTER - III
MAJOR-THEORY
DATA STRUCTURES

OBJECTIVES :

- To enable the students to understand the properties of various data structures .
- Identify the strength and weakness of different data structures
- Design and employ appropriate data structures for solving computing problems

HOURS / WEEK - 5		
HOURS / SEMESTER - 75		
CIA	CO-CURRICULAR ACTIVITIES	LECTURE
10	5	60

UNIT I (LECTURE HOURS: 10)

Data Structures -An Introduction

Concept of Data Structures, Types of Data Structures - Choice of data structures, Linear and Nonlinear

Arrays

Introduction, One Dimensional Array - Subscript of an array, Operations

Multi-Dimensional Arrays - Representation

Representation of Arrays in Physical Memory - Physical address computation of elements of Arrays

Applications of Arrays

Polynomial Representation - Polynomial Representation

Sparse Matrix Representation - Addition and Transpose

Tutorial - Arrays

Maximum Sum of Rows, Columns and Diagonals - Maximum Sum of Rows, Columns and Diagonals

UNIT II (LECTURE HOURS: 13)

Pointers

Introduction, Pointer Variables - The '&' operator and '*' operator, Dangling Pointers

More on Pointers

Pointers and Arrays, Array of Pointers - Pointers and Arrays, Array of Pointers

Pointers & Structures - Pointers & Structures

Dynamic Allocation - Dynamic Allocation, Self referential structures

Linked List

Introduction, Operations on linked list - Introduction, Create, Insert, Delete, Search, Traversal

Implementation

Creation - Creating a singly linked list

Insertion - Insertion in a singly linked list

Deletion - Deletion in a singly linked list

Variations of linked list

Circular linked list, Doubly linked list - Circular linked list, Doubly linked list

Tutorial - Linked List

Find every kth node in a linked list - Find every kth node in a linked list

UNIT III (LECTURE HOURS: 12)

Stack

Introduction - Introduction

Stack Operations - Push, Pop, Display

Implementation

Stack using Array - Stack using Array

Linked Stack - Stack using Linked List

Applications of Stacks

Arithmetic Expressions - Conversion of Infix to postfix expression

Evaluation of Expressions - Evaluation of Postfix Expressions

SEMESTER - III
MAJOR-THEORY
DATA STRUCTURES

Tutorial - Stack

Nearest Largest Number - Nearest Largest Number

UNIT IV (LECTURE HOURS: 11)

Queues

Introduction - Introduction

Queue Operations - Enqueue, Dequeue

Implementation

Queue using Array - Queue using Array

Linked Queue - Queue using Linked List

Types of Queue

Circular Queue - Circular Queue

Priority Queue - Priority Queue

Deque - Deque

Tutorial - Queue

Queue in a Hospital - Queue in a Hospital

UNIT V (LECTURE HOURS: 14)

Trees

Introduction - Introduction

Basic Terminologies - Basic Terminologies

Binary Trees

Introduction, Representation of Binary Tree - Types, Properties, Linear representation, Linked representation **Traversal of Binary Trees** - Inorder, Preorder ,Postorder

Types of Binary Trees

Expression Trees - Expression Trees

Binary Search Tree - Operations

Heap Tree - Representation, Operations

Threaded Binary Tree - Representation

Tutorial - Trees

Problems on Tree Expressions - Inorder, Preorder ,Postorder

Problems on Binary Search Tree - Create, Insert, Delete

Text Books :

Data Structures using C | Edition:2 | Pearson Education | A K Sharma(2013)

Reference Books :

Data Structures and Algorithms: Concepts, Techniques and Applications | Edition: | McGraw Hill Education (India) Pvt Ltd | G A V Pai ( )

SEMESTER - III
MAJOR-THEORY
SOFTWARE ENGINEERING

OBJECTIVES :

- Develop in students a critical understanding of issues related to the engineering of large complex software systems.
- Know how to develop a set of requirements,
- Apply rigorous software analysis.
- To design, code & test their work.

HOURS / WEEK - 5		
HOURS / SEMESTER - 75		
CIA	**CO-CURRICULAR ACTIVITIES**	**LECTURE**
10	**5**	**60**

UNIT I (LECTURE HOURS: 13)

The Evolving Role Of Software And Software

The Evolving Role Of Software - Explanation

Software: Software Characteristics and Applications - Definition, Explanation about Software Characteristics.

Software Crisis And Myths

Software Crisis - Explanation about crisis.

Software Myths - Definition, Explanation about Management myths, Customer Myths, Practitioner's Myths.

Software Engineering: A Layered Technology

Process, Methods, and Tools , A Generic View of Software Engineering - Explanation about Process, Methods and Tools, A Generic View of Software Engineering

The Software Process

The Software Process - Definition, Explanation about Process Maturity, diagram- A Common Process Framework.

Software Process Models And The Prototyping Model

Software Process Models - Definition, Explanation about Process Model, Diagram - The phases of a problem, The Phases within Phases.

The Prototyping Model - The Linear Sequential Model - Explanation, Diagram

The Rad Model And Evolutionary Software Process Models

The RAD Model - The Prototyping Model, The RAD Model-Explanation, Diagram.

The Incremental Model and Spiral Model - The Incremental Model, The Spiral Model - Explanation, Diagram.

UNIT II (LECTURE HOURS: 13)

Requirements Engineering

Requirements Elicitation, Requirements Analysis and Negotiation, Specifications, System Modeling, Validation, Management. - Explanation about Requirements Elicitation, Requirements Analysis and Negotiation, Specifications, System Modeling, Validation, Management.

System Modeling

System Modeling - Explanation, Diagram - Sample Model Template, System context Diagram, System Flow Diagram.

Requirements Analysis And Requirements Elicitation For Software

Requirements Analysis - Explanation, Diagram - Analysis as a bridge between system engineering and software design.

Initiating the process, Facilitated Application Specification Techniques, Quality Function Deployment, Use-cases - Explanation about Initiating the process, Facilitated Application Specification Techniques, Quality Function Deployment, Use-cases.

Software Prototyping And Specification

Selecting the Prototyping Approach, Prototyping Methods and tools. - Explanation about Selecting the Prototyping Approach, Prototyping Methods and tools.

Specification Principles, Representation, The Software Requirements Specification - Explanation about Specification Principles, Representation, The Software Requirements Specification.

The Elements Of Analysis Model And Data Modeling

The Elements of Analysis Model - Explanation about DFD, STD, Diagram - The Structure of the analysis Model.

SEMESTER - III
MAJOR-THEORY
SOFTWARE ENGINEERING

Data Modeling - Explanation about Data Objects, Attributes, and Relationships, Cardinality and Modality, Entity / Relationship Diagrams.

Functional Modeling And Information Flow

Functional Modeling And Information Flow - Explanation about Data Flow Diagrams, Extensions for Real- Time Systems, Ward and Mellor Extensions, Hatley and PribhaiExtensions.

The Mechanics Of Structured Analysis And The Data Dictionary

The Mechanics of Structured Analysis - Explanation about Creating an Entity / Relationship Diagram, Data Flow Model, Control Flow Model, Control Specification, Process Specification.

The Data Dictionary - Definition, Explanation about Data Dictionary.

UNIT III (LECTURE HOURS: 11)

Design For Object Oriented System And The System Design Process

Conventional Vs OO Approaches, Design Issues, The OOD Landscape, A Unified Approach to OOD - Explanation Diagram-The OOD design pyramid, Conventional Vs OO Approaches, Diagram - Translating OOA model into an OOD model, Design Issues, The OOD Landscape, A Unified Approach to OOD, Diagram - Process flow for OOD.

Partitioning the Analysis Model, Concurrency and Subsystem Allocation, The Task Management Component, User Interface Component, The Data Management Component, The Resource Management Component, Inter subsystem Communication - Explanation, Partitioning the Analysis Model, Concurrency and Subsystem Allocation, The Task Management Component, User Interface Component, The Data Management Component, The Resource Management Component, Inter subsystem Communication.

Software Design & Engineering And The Design Process & Principles

Software Design and Software Engineering - Explanation, Diagram-Translating the analysis model into a software design

The design process - Explanation, Explanation about Design and Software Quality, The Evalution of Software Design.

Design Principles - Explanation about Principles for software Design.

Design concepts

Design Concepts - Explanation, Explanation about Abstraction, Refinement, Modularity, Software Architecture, Control Hierarchy, Structural Partitioning, Data Structure, Software Procedure, Information Hiding.

Effective modular design and design heuristics for effective modularity

Effective Modular Design - Explanation, Explanation about Functional Independence, Cohesion, Coupling.

Design Heuristics for Effective Modularity - Explanation about set of Heuristics.

UNIT IV (LECTURE HOURS: 12)

Mapping requirements into a software architecture and Transform & transaction mapping

Transform flow, Transaction flow - Explanation, Explanation about Transform flow, Transaction flow.

Transform Mapping -An Example, Design Steps - Explanation, Explanation about An Example, Design Steps.

Transaction Mapping - An Example, Design Steps - Explanation, Explanation about An Example, Design Steps.

User Interface Design And Activities

Interface Design Models, The User Interface Design Process - Explanation about Interface Design Models, The User Interface Design Process.

Defining Interface Objects and Actions, Design Issues. - Explanation about Defining Interface Objects and Actions, Design Issues.

Software Testing Fundamentals, White Box Testing, Basis Path Testing, Control Structure Testing, And Black Box Testing

Software Testing Fundamentals - Testing Objectives, Principles, Testability - Explanation about Testing Fundamentals, Testing Objectives, Testing Principles, Testability

White Box Testing - Explanation about White box testing, The Nature of Software Defects.

Basis Path Testing - Flow Graph Notation, Cyclomatic Complexity, Deriving Test Cases, Graph Matrices - Explanation about Basic Path Testing, Flow Graph Notation, Cyclomatic Complexity, Deriving Test Cases, Graph Matrices, Flowchart.

Control Structure Testing - Condition Testing, Data Flow Testing, Loop Testing - Explanation, Explanation about Condition Testing, Data Flow Testing, Loop Testing.

Black Box Testing - Graph-Based Testing Methods, Equivalence Partitioning, Boundary Value Analysis, Comparison Testing, Orthogonal Array Testing - Explanation about Black Box Testing, Graph-Based Testing Methods, Equivalence Partitioning, Boundary Value Analysis, Comparison Testing, Orthogonal Array Testing.

UNIT V (LECTURE HOURS: 11)

A Strategic Approach To Software Testing, Validation Testing, System Testing, And The Art Of Debugging

SEMESTER - III
MAJOR-THEORY
SOFTWARE ENGINEERING

A Strategic Approach to Software Testing - Verification and Validation, Organizing for Software Testing, A Software Testing Strategy, Criteria for Completion of Testing - Explanation about Strategic Characteristics, Verification and Validation, Organizing for Software Testing, A Software Testing Strategy, Criteria for Completion of Testing

Validation Testing - Validation test Criteria, Configuration Review, Alpha and Beta testing. - Explanation about Validation Testing, Validation test Criteria, Configuration Review, Alpha and Beta testing.

System Testing - Recovery Testing, Security Testing, Stress Testing, Performance Testing - Explanation about System Testing, Recovery Testing, Security Testing, Stress Testing, Performance Testing.

The Art of Debugging - The Debugging Process, Psychological Considerations, Debugging Approaches. - Explanation about Debugging, The Debugging Process, Psychological Considerations, Debugging Approaches.

Software Quality

McCall's Quality Factors, FURPS, ISO 9126 Quality Factors, The Transition to a Quantitative view - Define Quality, Explanation about McCall's Quality Factors, FURPS, ISO 9126 Quality Factors, The Transition to a Quantitative view.

The Software Reengineering And Reverse Engineering

Software Reengineering -Software Maintenance, A Software Reengineering Process Model - Explanation about Software Maintenance, A Software Reengineering Process Model, and Diagram.

Reverse Engineering - Reverse Engineering to Understand Processing and Data, Reverse Engineering User Interfaces. - Explanation about Reverse Engineering Process and Diagram, Reverse Engineering to Understand Processing, Reverse Engineering to Understand Data, Reverse Engineering User Interfaces.

Building blocks for case and a taxonomy of case tools

Building Blocks for CASE - Explanation about CASE Building Blocks and Diagram, Diagram about Integration Options. **A Taxonomy of CASE Tools** - Concept, Function Classification about CASE Tools,

Text Books :

Software Engineering | Edition:5th Edition | TMH Publishers | Roger S Pressman(2001)

Reference Books :

A discipline for Software Engineering | Edition: | Pearson Education Publishers | Watts S Humphrey(2001)
Software Engineering | Edition:6th Edition | Pearson Education Publication | Ian Somerville(2001)

SEMESTER - III
MAJOR-THEORY
OPERATING SYSTEMS

OBJECTIVES :

- Understand the components, uses of Operating system and types of operating systems based on processing method.
- Identify memory management strategies.
- Compare and contrast virtual address and physical address.
- Determine the Scheduling algorithm using FCFS , SJF , and Round robin techniques.
- Define Synchronization using Banking , Peterson algorithm and Thundering herd problem .
- Determine the accuracy of Dining Philosopher problem and the Security policy model using Bell-la padula and Biba
- Categorize user level and kernel level threads

HOURS / WEEK - 5		
HOURS / SEMESTER - 75		
CIA	CO-CURRICULAR ACTIVITIES	LECTURE
10	5	60

UNIT I (LECTURE HOURS: 11)

INTRODUCTION TO OPERATING SYSTEMS

Computer System Organization - Computer System Operation|| Storage structure|| I/O Structure **Operating System Structure** - Simple Structure|| Layered Approach||Microkernel|| Modules **Computer System Architecture** - Single processor Systems , Multiprocessor Systems , Clustered Systems **OS Usage** - The Layers in Systems , Hard ware Abstraction, Resource Management **PC Hardware** - Memory Addresses , IO Addresses , Memory mapped IO addresses **General Purpose Registers** - 8088 , 80386 microprocessor , Backward Compatibility.

PROGRAM TO PROCESS

Executing Programs - Process

Process memory map - Stack , Heap , Data , Text

Beyond the programs memory map - Kernel

Communicating with OS - System call, System call vs procedure call

System call Interface - File System call

SHARING THE CPU

Multiprogramming - Idle CPU cycles , starvation

Multitasking - Time slice

Multiprocessors - Processor core , thread , Chip, Race condition

Synchronization - Lock, Scheduling

TUTORIAL : SHARING THE CPU

Multiprogramming , Multitasking - 1.App1 , App2 , App3 wants to access the CPU .App5 is running in the CPU. When the CPU is Idle? How the performance is increased.

UNIT II (LECTURE HOURS: 12)

INTRODUCTION TO PROCESSES

Virtual address map of a process - Process page table

Kernel - Where does kernel reside

Kernel and user space - Max limit , Max size

Contiguous mapping of kernel - Converting Virtual address to Physical address , Converting Physical address to Virtual address **Kernel and multiple processes** - Kernel about a data process

PROCESS CONCEPT AND OPERATIONS

The process - Process in memory

Process state - New ,Running , Waiting , Ready , Terminated

Process Control Block - Process state , Program counter , CPU registers, CPU scheduling information , Memory management information, Accounting information , I/O status information.

Threads - thread

Process Creation - A tree of process

Process Termination - Exit system call , wait system call

MEMORY MANAGEMENT

STRATEGIES

SEMESTER - III
MAJOR-THEORY
OPERATING SYSTEMS

Swapping - Swapping of two processes using a disk as a backing store
Contiguous Memory Allocation - Memory mapping and protection , Memory allocation , Fragmentation
Paging - Basic method, Hardware support , Protection ,Shared pages
VIRTUAL MEMORY
Demand paging - Basic concepts , performance of Demand paging
Memory Management Unit - Virtual address , Physical address
MMU Mapping - MMU Mapping 32 bit systems, Two level page translation
Working of Virtual memory - Page table , trap , Example.
Page Replacement - FIFO replacement , Optimal page replacement ,LRU page replacement
TUTORIAL: MEMORY MANAGEMENT STRATEGIES
Contiguous Memory Allocation - The size of the RAM is 180k. Process 1 and process 4 occupy the RAM. Process 1 has the size 60K.Size of Process 4 is 110K. Process 1 Completes its execution the area in RAM is deallocated. New process process 6 Has the size 65K cannot start ? Why? How to avoid underutilization of RAM.

UNIT III (LECTURE HOURS: 12)

PC BOOTING
Powering Up : Reset , BIOS - Reset , BIOS
Powering Up : MBR, Bootloader , OS - MBR, Bootloader, OS
PROCESS SCHEDULING - BASIC CONCEPTS
CPU- I/O Burst Cycle - Burst duration
CPU Scheduler - Short term scheduler
Preemptive Scheduling - Pre- emptive , Non- preemptive
Dispatcher - Dispatch latency
Scheduling criteria - CPU Utilization , Through put , Turn- around time , waiting time , Response time
SCHEDULING ALGORITHMS
First Come , First Served Scheduling - Average Waiting Time , Turn around time **Shortest Job First Scheduling** -

Average Waiting Time , Turn around time

Priority Scheduling - Starvation
Round Robin Scheduling - Time quantum , Processor sharing
Multilevel Queue Scheduling - Multilevel Queue Scheduling
Multilevel Feedback Queue Scheduling - Multilevel Feedback Queue
INTERRUPT
Hardware Interrupt - Programmable Interrupt controller
Interrupt Handling - What more happens when there is an interrupt
Software Interrupts - Explanation , Example
CPU Context switching - Context switch, Timer Interrupt ,Process context, Context switch overhead
MULTITHREADED PROGRAMMING
Motivation - Single threaded multi threaded process
Benefits - Responsiveness , Resource sharing , Economy , scalability
Multicore programming - Dividing activities , Balance , Data splitting , Data dependency, Testing and debugging
TUTORIAL : SCHEDULING ALGORITHMS
Shortest Job First Scheduling , Priority Scheduling , Round Robin Scheduling - Jobs A, B, C, D and E with running time of 10, 7, 2, 6 and 8 respectively arrive at a computer in the time order 0, 2, 5, 7, and 9. If the priority of these jobs is in the order C, B, E, A, D and the time slice is 3 units of time. Compute the time required to execute jobs A through E .

UNIT IV (LECTURE HOURS: 13)

MULTIPLE- PROCESSOR SCHEDULING
Approaches to Multiple-Processor Scheduling - Symmetric multiprocessing , Asymmetric multiprocessing
Processor Affinity - Soft affinity , Hard affinity
Load Balancing - Push migration , pull migration
Multicore Processors - Multi threaded multi core systems
INTER PROCESS COMMUNICATION
Shared Memory - Shared memory in Linux
Message passing - Send , receive
SYNCHRONIZATION
Motivating scenario - Shared variable
Race condition - Critical section

SEMESTER - III
MAJOR-THEORY
OPERATING SYSTEMS

SOFTWARE SOLUTION FOR CRITICAL SECTION

Using Interrupts - Preventing Context switching

Dead lock - Context switch

Peterson's solution - Favored process

Bakery algorithm - Simplified Bakery algorithm , Original Bakery Algorithm

Mutex - Sleep , Wakeup

Thundering herd problem - Context switch , starvation, Sleep, Wakeup

SEMAPHORES

Usage , Implementation - Producer Consumer problem

Deadlocks and Starvation , Priority Inversion - Producer Consumer problem

TUTORIAL : SOFTWARE SOLUTION FOR CRITICAL SECTION

Peterson's solution , Bakery algorithm - Consider a system consisting of processes. Each process has a segment of code, called a critical section (CS) .Several processes access and manipulate the same data, how to prevent the race condition.

UNIT V (LECTURE HOURS: 12)

CLASSIC PROBLEM OF SYNCHRONIZATION

Bounded Buffer Problem - First try, Second try, Solution using mutex , solution using semaphores **Readers-Writers Problem** - Mutual exclusion , Hold and wait, No preemption , Circular wait.

The Dining Philosophers problem - Deadlock state , No dead lock

DEADLOCK AVOIDANCE

Safe state , Unsafe state - State matrix

Banker's algorithm - Example

DEAD LOCK PREVENTION

Denying the wait for condition - Resource utilization

Denying the " No preemption" condition - Havender's second strategy

Denying the " Circular wait" condition - Linear ordering

SECURITY

Security Goals - Secrecy , Integrity , Availability.

Access control systems - Security Policy , Security Model , Security Mechanism

Direct Access control - Access matrix model

Mandatory access control - Classification level , clearance level

Bella Padula Model - No read up , No write down.

Biba Model - No read down , No write Up

TUTORIAL : DEAD LOCK AVOIDANCE , DEADLOCK PREVENTION

Safe state, Unsafe state, Banker's algorithm, Denying the wait for condition, Denying the No pre-emption condition , denying circular wait condition - A system has four processes P1 through P4 and two resource types R1 and R2. It has 2 units of R1 and 3 units of R2. Given that: P1 requests 2 units of R2 and 1 unit of R1 P2 holds 2 units of R1 and 1 unit of R2 P3 holds 1 unit of R2 P4 requests 1 unit of R1 Show the resource graph for this state of the system. Is the system in deadlock, and if so, which processes are involved?

Text Books :

Operating system concepts | Edition:Eighth | Wiley publication | ABRAHAM SILBERSCHATZ AND GREG GAGNE AND PETER BAER GALVIN ( )

Reference Books :

Operating Systems Internals and Design Principles | Edition:Seventh | Pearson Publication | WILLIAM STALLINGS ( )

SEMESTER - III
MAJOR-THEORY
FUNDAMENTALS OF NETWORKS

OBJECTIVES :

- Understanding the Basic Concepts of Networks
- Understanding the different Types of Cables, Connectors, and Networking Devices.
- Getting familiarization with Ethernet and IP Address Configurations .

HOURS / WEEK - 5		
HOURS / SEMESTER - 75		
CIA	**CO-CURRICULAR ACTIVITIES**	**LECTURE**
10	**5**	**60**

UNIT I (LECTURE HOURS: 12)

Networking Overview

Introduction to Networking, Network Types - LAN,WAN,MAN

Networking Terms, Networking Facts,Practice Questions - Transmission Media, Network interfaces, Protocols

Networking Topologies

Network Topologies,Topology Facts , Practice Questions - Bus, Ring, Star, Mesh, Terminators

The OSI Model

The OSI Model,The OSI Model Facts - Provides a common language and reference points for network professionals, Aids in troubleshooting, Provides modularity.

The OSI Model Layers,The OSI Model Communications - HTTP, Telnet, FTP, TFTP, SNMP, Formats or "Presents" the data

OSI Layers Facts,Practice Questions - Application, Presentation, Session, Transport, Network, Data Link, Physical.

Network Signaling

Network Signalling,Transmission Systems - What is the purpose of Network signalling, Difference between baseband and broadband.

Network Signalling Facts,Practice Questions - Return-to-Zero, Non-Return-to-Zero, Manchester, De-Multiplexing.

Network Protocols

TCP/IP Protocol Suite,Common Network Services - How does TCP differ from UDP, How does a protocol suite differ from a protocol.

Exploring Network Services,Common TCP/IP Protocols,Practice Questions - Web Services, HTTP, HTTP over SSL, SSL, Security Protocols, TLS, File Transfer, POP3, IMAP, IMAP4

Numbering Systems

Numbering System,Numbering System Facts,Practice Questions - What are the possible values in a binary number, Difference between a binary number and a hexadecimal numbering system , Binary (Base 2), Octal (Base 8), Hexadecimal (Base 16)

UNIT II (LECTURE HOURS: 12)

Twisted Pair

Twisted Pair,Twisted Pair Facts - Why are wires twisted together in twisted pair cables, Difference in CAT3 and CAT5 cables, CAT 3, CAT 5E, CAT 6, CAT 6A.

Connect to an Ethernet Network,Practice Questions - Ethernet LAN using Twisted Pair cables

Coaxial

Coaxial,Coaxial Cable Facts,Connect to Cable Modem,Practice Questions - What is the function of the wire mesh in coaxial cables, which part of the cable is used to carry data, RG58, RG59, RG6, BNC

Fiber Optic

Fiber Optic,Fiber Optic Facts - Totally immune to EMI, Very expensive, Highly resistant to eavesdropping, Difficult to work with.

Connect Fiber optic cables,Practice Questions - Connector A (red) is Tx, Connector B (black) is Rx.

Wiring Implementation

Twisted pair cable construction,Cable construction facts - Straight-through, Crossover, What is a patch panel used for.

SEMESTER - III
MAJOR-THEORY
FUNDAMENTALS OF NETWORKS

Wiring Distribution,Using Punchdown blocks - Using Punchdown
Wiring Distribution Facts,Connect Patch Panel Cables 1 - Demarcation point, Main Distribution Frame, Demarc extension, Horizontal cross connect, vertical cross connect
Connect Patch Panel Cables 2, Practice Questions - Use port IT Adm on the patch panel
Troubleshooting Network Media
Troubleshooting Copper Wires Issues, Copper Wiring Troubleshooting facts - EMI and RFI, Crosstalk, Near End Crosstalk, Far end Crosstalk.
Troubleshooting Fiber opting Wiring Issues, Fiber Optic Wiring Troubleshooting facts - Connectors, Polishing, Cabling, Media adapters, Attenuations
Troubleshooting tools, Troubleshooting tool facts, Practice Questions - Loopback plug, Smart jack, cable tester, time-domain reflectometer

UNIT III (LECTURE HOURS: 12)

Network Adapters
Network Adapters - Connect a media convertors, What does FF-FF-FF-FF-FF-FF indicates **Network**
Adapter Facts - A transceiver, Demodulation, Transceiver modular, Media convertor
Select and install a network adapter - Install a NIC
Connect a media converter - Port 23 is Tx, Port 24 is Rx.
Practice Questions - Practice questions.
Network Devices
Network Devices - At which OSI model layer do wireless access points operate, what type of device do you use to translate from one network architecture to another
Network Connection Device Facts - Hub, Bridge, Switch, Wireless AP
Install a Hub,Select a Networking Device - Home-PC, Use AC-to-DC power adapter
Practice Questions - Practice Questions
Internetwork Devices
Internetwork Devices, Internetwork Devices Facts - Router, Firewall, Layer 3 switch
Select a Router - SOHO network & Internet, Select correct cables.
Practice Questions - Practice Questions

UNIT IV (LECTURE HOURS: 12)

Ethernet
Ethernet Architecture - What logical topologies are supported on an Ethernet network, what device is used to enable full duplex, what is the purpose of the backoff on Ethernet network .
Ethernet Facts - Topology, Networking Devices, Transmission Media, Media Access Method.
Practice Questions - Practice Questions
Ethernet Specifications
Ethernet Specifications, Reconnect to an Ethernet Network,Practice Questions - Reconnect to an Ethernet Network, Ethernet, 10BaseT, 10BaseFL, 100BaseTX.
Connecting Network Devices
Connecting Devices, Device Connection Facts - Domain 3.0 Ethernet, When would you use a rollover cable? **Connect Network Devices, Practice Questions** - Practice questions
Troubleshooting Physical Connectivity
Troubleshooting the Fault Domain - Explore physical connectivity.
Fault Domain Trobleshooting Facts,Troubleshooting the Link Status - Bus, Star, Ring, Mesh
Link Status Troubleshooting facts,Exploring Physical Connectivity - Unlit, Activity, Collision, Red/Amber, Solid Green.
Trouble Shooting Physical Connectivity 1,Trouble Shooting Physical Connectivity2 - Network technician for a small corporate network.
Trouble Shooting Physical Connectivity 3,Trouble Shooting Physical Connectivity 4 - Physical Connectivity 3 **Practice Questions** - Practice Questions

UNIT V (LECTURE HOURS: 12)

IP Addressing
IP Addressing,IP Address Facts - What is an Octet, Configure IP address, How does VLSM works, IP address is a 32-bit binary numbers.
Subnetting ,Subnetting Facts - Increase the number of devices, Decrease the number of devices, subnetting uses custom subnet masks.

SEMESTER - III
MAJOR-THEORY
FUNDAMENTALS OF NETWORKS

Variable Length Subnetting facts - Class A, Class B, Class C, classless addresses, Partial Subnetting, VLSM. **IP Address Assignment, Configure IP Address** - Subnet, Host address, Default Gateway, TCP/IP settings **Configure IP Address on Mobile Devices** - Insecure port 143, IMAP mail server
IP Addressing Facts - DHCP, Static (manual) assignment.
Practice Questions - Practice Questions
Alternate IP Addressing
APIPA - Configure an alternate IP configuration.
Configuring Alternate Addressing - IP and DNS addresses automatically
Configure Alternate Addressing - Copy the static configuration settings
Alternate IP Addressing facts - APIPA, Alternate IP Configuration.
Practice questions. - Practice questions.

Text Books :
CCENT/CCNA ICND1 100-105, Official Cert Guide Wendell Odom, CCIE NO.1624 | Edition:ICND1 100-105 | Ciscopress.com | WendellOdom(2016)

Reference Books :
Computer Networks | Edition:5th | Prentice Hall | Andrew S. Tanenbaum AND David J.Wetherall(2014)

SEMESTER - III
MAJOR-PRACTICAL
PROGRAMMING LAB -III(DATA STRUCTURES & FUNDAMENTALS OF NETWORKS)

OBJECTIVES :

- To enable the students to understand the properties of various data structures .
- Design and employ appropriate data structures for solving computing problems.

HOURS / WEEK - 5		
HOURS / SEMESTER - 75		
CIA	**CO-CURRICULAR ACTIVITIES**	**LECTURE**
10	**5**	**60**

UNIT I (LECTURE HOURS: 12)

ARRAYS

Write a C program that finds the Second largest element in a given list of N numbers. - Aim ,Algorithm **Write a C Program to perform Matrix operations** - Aim ,Algorithm

Linked List

Write a C program for creating a linked list and perform insertion of a node into the list - Aim ,Algorithm **Write a C program for creating a linked list and perform deletion of a node from the list** - Aim ,Algorithm

Given a singly linked list, write a C program to remove every Kth node. The task is to complete a method deleteK that takes two argument, head of linked list and an integer k. The method returns the head of the new linked list

- Aim ,Algorithm

UNIT II (LECTURE HOURS: 12)

Stack

Implement the concept of stack as using array, such that an array named as item is used to store the elements of the stack and top is used to store the position of the top most element of the stack which is initialized to -1.Use Push () function to insert an element to the stack and Pop () function to remove an element from the stack. Use Display () of function to display all the elements of the stack - Aim ,Algorithm

Implement the concept of stack as using Linked List, such that struct node contains two elements named data and next where data contains the information and next contains the address of next node. Use Push () function to insert an element to the stack and Pop () function to remove an element from the stack. Use Display () of function to display all the elements of the stack - Aim ,Algorithm

Given an array, print the Next Greater Element (NGE) for every element. The Next greater Element for an element x is the first greater element on the right side of x in array. Elements for which no greater element exist, consider next greater element as -1. Write a stack based program to find the next greater element. - Aim ,Algorithm

UNIT III (LECTURE HOURS: 12)

Queue

Implement the concept of queue using array, such that an array named as data is used to store the elements of the queue and front is used to store the position of the front element of the queue and rear is used to store the last element of the queue. Use insertQ() function to insert an element to the queue and deleteQ () function to remove an element from the queue. Use DisplayQ () of function to display all the elements of the Queue. - Aim ,Algorithm

Implement the concept of queue using linked list, such that an array named as data is used to store the elements of the queue and front is used to store the position of the front element of the queue and rear is used to store the last element of the queue. Use insertQ() function to insert an element to the queue and deleteQ () function to remove an element from the queue. Use DisplayQ () of function to display all the elements of the

Queue - Implement the concept of queue using linked list, such that an array named as data is used to store the elements of the queue and front is used to store the position of the front element of the queue and rear is used to store the last element of the queue. Use insertQ() function to insert an element to the queue and deleteQ () function to remove an element from the queue. Use DisplayQ () of function to display all the elements of the Queue

SEMESTER - III

MAJOR-PRACTICAL

PROGRAMMING LAB -III(DATA STRUCTURES & FUNDAMENTALS OF NETWORKS)

Implement the concept of circular queue using array, such that an array named as data is used to store the elements of the circular queue and front is used to store the position of the front element of the queue and rear is used to store the last element of the queue. Use insertCQ () function to insert an element to the queue and deleteCQ () function to remove an element from the queue. Use DisplayCQ () of function to display all the elements of the Queue. - Aim ,Algorithm

UNIT IV (LECTURE HOURS: 12)

PROGRAM 1

Many network problems can be fixed at the Physical layer of a network. For this reason, it is important to have a clear understanding of which cables to use for your network connections. At the Physical layer (Layer 1) of the OSI model, end devices must be connected by media (cables). The type of media required depends on the type of device being connected. In the basic portion of this lab, straight-through or patch-cables will be used to connect workstations and switches. In addition, two or more devices communicate through an address. The Network layer (Layer 3) requires a unique address (also know as a logical address or IP Addresses), which allows the data to reach the appropriate destination device. Addressing for this lab will be applied to the workstations and will be used to enable communication between the devices. - Task 1: Create a Peer-to-Peer Network. || Task 2: Identify the Cables used in a Network || Task 3: Cable the Peer-to-peer Network || Task 4: Connect Your Workstations to the Classroom Lab Switch || Task 5: Reflection

PROGRAM 2

When working in Packet Tracer, in a lab environment, or in a corporate setting it is important to know how to select the proper cable and how to properly connect devices. This activity will examine device configurations in Packet Tracer, select the proper cable based on the configuration, and connect the devices. This activity will also explore the physical view of the network in Packet Tracer. - Task 1: Connect the Devices in the Standard Lab Setup || Task 2: View the Standard Lab Setup in the Physical Workspace .

UNIT V (LECTURE HOURS: 12)

PROGRAM 3

You have been asked to implement the standard lab topology, but with a new IP addressing scheme. You will use many of the skills you have learned to this point in the course. - Task 1: IP Subnet Planning || Task 2: Finish Building the Network in Packet Tracer || Task 3: Configure the Network || Task 4: Test the Network.

PROGRAM 4

You have been asked to repair some problems in the network model related to the Ethernet LAN connected to R2-Central - Task 1: IP Subnet Planning || Task 2: Repair Problems with the Ethernet Switched LAN || Task 3: Test the Network.

Text Books :

"Data Structures using C" | Edition:2nd Edition | Pearson Education | A KSharma(2013)
CENT/CCNA , Official Cert Guide, CCIE NO.1624 | Edition:ICND1 100-105 | Ciscopress.com | Wendell Odom(2016)

SEMESTER - III
ALLIED-THEORY
OPERATIONS RESEARCH FOR INFORMATION TECHNOLOGY

OBJECTIVES :

On successful completion of this course the students should gain knowledge about the lpp , optimal use of resourse techniques

HOURS / WEEK - 5		
HOURS / SEMESTER - 75		
CIA	**CO-CURRICULAR ACTIVITIES**	**LECTURE**
10	**5**	**60**

UNIT I (LECTURE HOURS: 12)

Linear Programming

Introduction - Definition
Formulation of Linear programming Problems - Procedure and related problems
Limitations of Linear Programming - Theoretical explanation
Graphical solution to LPP - Graphical problems
Simplex Method - Related problems
Artificial Variable Technique (BIG M Method) - Related problems

UNIT II (LECTURE HOURS: 12)

Inventory control

Introduction - Explanation of inventory
Inventory cost - Various symbols
Purchase model with no Shortages - Related problems
Purchasing with Shortages - Related problems
Manufacturing model with no Shortages - Related problems

Transportation model

Introduction - Basic definitions
NWC Rule - Related problems
LCM Rule - Related problems
VAM - Related problems

UNIT III (LECTURE HOURS: 12)

Assignment Model

Introduction - Basic definition
Mathematical formulation of the Problem - Related problems
Solution of an Assignment Problem - Related problems
Multiple Solution - Related problems
Hungarian Algorithm & Maximization in Assignment model - Related problems
Impossible Assignment - Related problems

Simulation

Introduction - Basic definition
Types of Simulation - Various types
Random Phenomena in Simulation - Related problems
Random Numbers - Related problems
Monte Carlo Simulation - Related problems

UNIT IV (LECTURE HOURS: 12)

PERT / CPM

Introduction - Basic definition
Concept of network - Procedure

SEMESTER - III
ALLIED-THEORY
OPERATIONS RESEARCH FOR INFORMATION TECHNOLOGY

Dummy Activities - Definition of activities
Critical Path - Related problems
Pert Model - Related problems
difference between PERT and CPM - Basic differences between pert & cpm

UNIT V (LECTURE HOURS: 12)

Sequencing problems
Introduction - Basic rules
Processing n jobs through two machines - Related problems
Processing n jobs through three machines - Related problems
Game theory
Introduction - Basic definition
Method of solving game theory problems - Related problems
Games with mixed Strategies - Related problems
Game with Dominance - Related problems

Text Books :
T1.Operations Research | Edition:1 | Keerthi Publishing House Pvt.Ltd | S.Dharani Venkatakrishnan(1990)

Reference Books :
R1.Resource management techniques | Edition:4 | A.R.Publications | V.Sundaresan,K.S.Ganapathy Subramanian & K.Ganesan(2007)

SEMESTER - IV
MAJOR-THEORY
DESIGN AND ANALYSIS OF ALGORITHMS

OBJECTIVES :

- Understand the time complexity of various data structures
- Selecting suitable data structures for any given problem
- Design Optimized algorithm and implement them for a given problem

HOURS / WEEK - 5		
HOURS / SEMESTER - 75		
CIA	**CO-CURRICULAR ACTIVITIES**	**LECTURE**
10	**5**	**60**

UNIT-I (LECTURE HOURS: 12)

ALGORITHM

Introduction - What is Algorithm? How does it look like?, Example, Why Algorithms?

Design of Algorithm - What is design of an algorithm?, Iterative and Recursive algorithms, The design steps, The design techniques, The role of Data structures in algorithm design.

Analysis of Algorithms - What is Analysis?, Why Analysis?

ASYMPTOTIC COMPLEXITY

How to do Analysis? - Definition, Big 'O' notation

Efficiency and Running time an Algorithm - Measuring the running time of an algorithm, How to find an efficiency of an algorithm

Complexity - Definition, Space and Time complexity, Difference between performance and complexity, Complexity classes **Worst case of an algorithm** - What is worst case of an algorithm?, How to determine complexity?

Calculating complexity with iterative version - statements, if statements, single, double and triple loop (for) examples **Calculating complexity with Recursive version** - While loop, Towers of Hanoi puzzle

The two basic data structure

Arrays and Lists - Introduction, storage of values, what is Array and List in storage point of view?

Array - Access an element with offset, Insert an element, the worst case input, Delete an element

List - Access an element, Inserting an element and Deleting an element using plumbing

Array vs. List - Difference between Array and List in complexity point of view

Analysis of Stack and Queue

Implementing stack using Array and List - Create, Inserting an element, Deleting an element, Traversing the data structure

Implementing queue using Array and List - Create, Inserting an element, Deleting an element, Traversing the data structure

UNIT-II (LECTURE HOURS: 12)

Hash Table

Hash Table - Introduction, Storing values in Hash Table, Hashing

Hash function - Choosing a good Hash function, Perfect Hash function, Load Factor

Basic Operations and their Time complexity - Search, Insert, Delete (Programs}

Collision Resolution techniques - Linear Probing ,Separate Chaining, Applications of Hash

Applications - Hash applications

Heap

Heap Structure - Introduction, Heapify and it's analysis

Types of heap - Max Heap, Min Heap

Max Heap and it's Analysis - Find max, insert, delete

Min Heap and it's Analysis - Find min, insert, delete

Other Heap operations - Merge, Meld, shift up and shift down

UNIT-III (LECTURE HOURS: 12)

SEARCHING

SEMESTER - IV
MAJOR-THEORY
DESIGN AND ANALYSIS OF ALGORITHMS

Search Problem - Definition
The Unsorted case - Linear Search : example, worst case
The sorted case - Binary Search : example, worst case
Comparision - Sorted Vs. Unsorted case
Sorting
Introduction - Definition, The need for sorting, Advantages
Selection sort - sorting using swap
Analysis - Recursive and Iterative versions
Insertion Sort - The swap example
Analysis - Recursive and Iterative versions
Comparision of sequential sorting - Selection sort Vs. Insertion Sort Vs. Bubble Sort
Divide and Conquer sorting
Divide and Conquer - Introduction
Merge Sort - Example
Analysis - Merge, Mergesort, Shortcomings
Quick Sort - Introduction, Sort with Pivot, Tony Hoar's actual partioning method
Quick Sort-Analysis - Randomization, Applications
STABLE VS. UNSTABLE SORT
Stable Sort - Definition, Example
Unstable Sort - Definition, Example
Which sorting is best? - Comparision

UNIT-IV (LECTURE HOURS: 12)

Binary Search Tree
operations - The structure of a BSTree, Find(v), Minimum, Maximum, Predecessor, Successor, insert, delete **Analysis** - Analysis of all operations
Graphs
Formal Representation - Directed Edge, Undirected Edge
Working with Graphs : Representation - Adjacency Matrix, Adjacency List
Graph Explorations
Breadth First Search(BFS) - Example, Algorithm
Analysis - Adjacency Matrix, Adjacency List
Depth First Search (DFS) - Example, Algorithm
DFS-Analysis - Adjacency Matrix, Adjacency List
Applications of BFS and DFS - Properties of DFS, Identifying connected components, Identifying cycles

UNIT-V (LECTURE HOURS: 12)

DAG (Directed Acyclic Graph)
Introduction - Directed Cycles, DAG definition, Features of DAG, Example
Greedy Algorithms
Greedy Method - Introduction, Example
SHORTEST PATH
Weighted Graphs - Overview
Shortest Path - Weighted Graph, classification
Single source shortest path - Dijksta's Algorithm, Example, Analysis
SPANNING TREES
Tree, Spanning Tree - Definition with example
Spanning Tree with costs - Example
Facts about Trees - Proof 1,2,3
Minimum Cost Spanning Trees - Prim's Algorithm, Analysis, Kruskal's Algorithm, Analysis
P & NP
The NP Class - Why NP?, What are P?, is P=NP?, NP Hard, NP Complete

SEMESTER - IV
MAJOR-THEORY
DESIGN AND ANALYSIS OF ALGORITHMS

Reference Books :

Grokking Algorithms - An Illustrated guide for programmers andother curious people | Edition:2016 | ManningPublications | Aditya Y. Bhargava( )

SEMESTER - IV
MAJOR-THEORY
BASIC IP SERVICES

OBJECTIVES :

·Enable the students to learn the basic principles and techniquesof IP services .
·To understand the concept of basic IP Configurations, switch management, routing, firewalls, WAN, network security and management.
·To understand the concept of networking and able to configure , verify, and troubleshoot complex computer networks.

HOURS / WEEK - 6		
HOURS / SEMESTER - 90		
CIA	**CO-CURRICULAR ACTIVITIES**	**LECTURE**
12	6	72

UNIT-I (LECTURE HOURS: 14)

DHCP Server Configuration & DHCP Relay & Multicast

Configuring a DHCP Server - Scope Name, Address Range, Subnet Mask
Configure a DHCP Server - Exclusions and Delays, Lease Durations, Scope Option for the Router, WINS Server. **Configuring a DHCP Options** - Configure Subnet1 scope as follows, DNS Servers as 163.128.78.93, Domain Name as CorpNet.com
Configure a DHCP Options - Configure a scope on the CorpDHCP12 server.
Create DHCP Exclusions - Create a range to exclude addresses 192.168.0.1 to 192.168.0.6 **Create DHCP Client Reservations** - Reservation Name, MAC address, IP Address. **Configure Host Addressing** - Configure Host Addressing
Configure a DHCP Client - Record the laptop's static IP, create an alternate TCP/IP connection **DHCP Configuration Facts** - DHCP Discover, DHCP Offer, DHCP Ack, DHCP Request **Practice Questions** - Practice questions.
Configuring DHCP Relay - Difference between RFC 1542 compliant router and a DHCP relay agent, Configure a DHCP Relay Agent
DHCP Relay Facts - RFC 1542 Compliant Router, DHCP Relay Agent
Configure a DHCP Relay Agents - Add the DHCP Relay Agent Routing Protocol, Add vEthernet(External)
Add a DHCP Server to an Another Subnet - ipconfig /all command, ipconfig /renew command
Practice Quesions - Practice quesions.
Multicast - Creates logical group of hosts.
Multicast facts - Unicasting, Broadcasting, Internet Group Management Protocol **Practice Questions** - Practice questions

Troubleshooting IP Configuration Issues & Troubleshooting IP Communication

IP Configuration Troubleshooting
Using ipconfig
Using ifconfig
IPconfig Utility Facts - Static IP Configuration, DHCP Configuration, Rogue DHCP Server **Explore IP Configuration** - ping, ipconfig or Tracert command utility
Troubleshooting IP Configuration Problem 1 - Your task in the lab, Network and Sharing Center
Troubleshooting IP Configuration Problem 2 - Your task in the lab, Network and Sharing Center
Troubleshooting IP Configuration Problem 3 - Your task in the lab, Network and Sharing Center
Troubleshooting IP Configuration Problem 4 - Your task in the lab, Network and Sharing Center **Practice Questions** - Practice questions.
Network Communication Troubleshooting - Difference between netstat and nbtstat commands, Information about IP configuration settings
Using ping and Tracet - Ping and Tracet
Network Communication Troubleshooting facts - Workstation A can't communicate with Workstation C **Using ARP, NETSTAT and NBTSTAT facts** - arp-a, netstat, netstat-a, netstat-r, netstat-s, nbtstat-c **Exploring Networking Communications** - PING OR TRACERT
Practice Questions - Practice questions

SEMESTER - IV
MAJOR-THEORY
BASIC IP SERVICES

Troubleshooting Name resolution

Name Resolution Troubleshooting - what are symptoms of name resolution problems?
Name Resolution Troubleshooting Facts - PING, TRACERT, OR TRACEROUTE, NSLOOKUP, DIG, HOST **Using nslookup** - Nslookup
Explore nslookup - use nslookup to query DNS for th4e CorpWeb server using its full name **Practice Questions** - Practice questions

UNIT-II (LECTURE HOURS: 16)

Switch Access

Device Access - Device Access
Using the command Line Interface(CLI) - managed and unmanaged switches
Device Connection facts - in-band management, out-of-band management
Password Levels - privilege levels
Configuring Line level passwords - console line password **Configuring Enable Mode Passwords** - enable password, enable secret
Modify System Passwords - MF5, hashing algorithm, four VTY lines
Configuring AAA Authentication - AAA authentication, TACACS server
Switch Password Facts - console, VTY, EXEC mode **Practice Questions** - Practice Questions

Switch IP Configuration

IP Address and Default Gateway Configuration - Why would you configure an IP address on a switch?
Switch IP Configuration Facts - switch# config terminal, switch(config)#interface vlan1
Configure Management VLAN Settings - Configure the VLAN1 interface, show run command
Configure Switch IP Settings - startup-config file
Practice Questions - Practice Questions

Switch Interface Configuration

Switching Operations - configure switch ports
Switch Forwarding Facts - flooding the frame, unicast address, forwarding the frame, filtering the frame **Switch Configuration Overview** - switch configuration
Switch Configuration Mode Facts - user exec, privileged exec, vlan database, global configuration **Switch Configuration Command List** - interface FastEthernet 0/14, interface GigabitEthernet 0/1 **Configure Switch Ports** - Fa0/4 and Fa0/5, Fa0/8 and Fa0/23 **Practice Questions** - Practice Questions

Virtual LANs

VLAN Overview - What are the two advantages to creating VLANs on your network? **VLAN Facts** - Fast Ethernet ports 0/1 and 0/2 are members of VLAN1 **Configuring VLANs** - VLAN global configuration commands **VLAN Command List** - vlan[1-4094],name[unique_name]
Create VLANs - n/a, accounting, research, unused **Exploring VLANs** - ping 12.0.0.2, ping12.0.0.3, ping 12.0.0.4 **Practice Questions** - Practice Questions

Trunking

Access and Trunk Ports - what is trunking, configure trunking
Configuring Trunking - configure the native Vlan
Trunking Facts - trunk line, access ports are connected to endpoint devices
Trunking Command List - Switchport mode trunk, Switchport mode access
Configure Trunking - Manually designate the following ports **Configuring the Native VLAN** - native VLAN
Configure the Native VLAN - show interfaces trunk and change the native VLAN from its default to VLAN
Configure Allowed VLANs - show run or show interfaces trunk **Practice Questions** - Practice Questions

Spanning Tree Protocol

Spanning Tree Protocol - why will a tie breaker never be necessary for the roots switch selection? when would you modify an STP mode?
Configuring STP - Configure STP
Selecting a Root Bridge - select a root bridge

SEMESTER - IV
MAJOR-THEORY
BASIC IP SERVICES

STP Facts - Switching loops, spanning tree Algorithm
Configure the Root Bridge - show spanning-tree vlan 1
Configure Rapid PVST+ - configure Rapid PSVT+ on each switch
Find STP Info
Configuring EtherChannels - Configure EtherChannels
EtherChannel Facts - Port Aggregation protocol, Link Aggregation control protocol
Practice Questions - Practice Questions
Switch Troubleshooting
Switch Troubleshooting - Duplex mismatch, broadcast storm
Switch Troubleshooting Facts - Collisions, Duplex mismatch, slow link speed
Practice Questions - Practice Questions
Routing Basics & Routing Protocols
Routing - Difference between static and dynamic routing, what is convergence **Routing**
Facts - routing table, loopback interface **Practice Questions** - Practice Questions
Routing Protocols Characteristics - Configure a router with static routes, enable OSPF routing
Routing Protocols Characteristics Facts - Scope, Metric, routing update method, classful or classless
Routing Protocols - Routing protocols
Routing Protocols Facts - RIP, EIGRP, OSPF, IS-IS, BGP
Configuring Routing - Configuring Routing
Configure Static Routes - Show cdp neighbors detail
Enable OSPF Routing
Practice Questions - Practice Questions
Network Address Translation
Network Address Translation - Implement NAT, configure internet connection sharing
Configuring NAT from the CLI - Configure NAT from the CLI
Configuring NAT on the NSA - Configure NAT on the NSA
Configuring Port Forwarding - Configure port forwarding
NAT Facts - Dynamic NAT, static NAT, dynamic and static
Practice Questions - Practice Questions
Routing Optimization & Routing Troubleshooting
Administrative Distance - what is administrative distance
Route Summarization - Benefits of Route summarization
High Availability
Routing Optimization Facts - Administrative distance values, route summarization, redundant default gateway routers **Practice Questions** - Practice Questions
Routing Troubleshooting - Hosts on a subnet to be configured, format for the default route entry in a routing table
Troubleshooting Routing
Troubleshooting Routing Facts - Can't access hosts outside the local subnet, can't communicate with any host on a specific network
Find Path Information 1
Find Path Information 2
Practice Questions - Practice Questions

UNIT-III (LECTURE HOURS: 14)

Firewalls
Firewalls
Firewalls Facts - Given a scenario and a Windows system, configure a basic host firewall
Common Ports - network-based firewalls, host-based firewall, access control lists
Configuring Windows Firewall - ICANN, well known ports, registered ports, dynamic
Configure a Host Firewall - configure windows firewall
Practice Questions - Choose an appropriate router which will provide security and the fastest local connection, Connect the router to both computers and to the dorm Internet connection, request new TCP/IP information from the router on Dorm-PC and Dorm-PC2, Configure the Windows Firewall on both computers
Security Appliances
All-In-One Security Appliances - Practice questions

SEMESTER - IV
MAJOR-THEORY
BASIC IP SERVICES

Security Solution Facts - Configure network security appliance access.
Configuring Network Security Appliance Access - Unified threat security devices or web security gateways **Configure Network Security Appliance Access** - Create a new Administrative User, Idle Timeout 15 Mins, User Type Administrator
Practice Questions - Practice questions
Firewall Design & Implementation
Firewall Network Design Principles - How do firewalls manage incoming and outgoing traffic, what does deny any statement do
Configuring a Perimeter Firewall - Demo: Configuring a Network Firewall
Firewall ACLs - Firewall ACLs
Creating Firewall ACLs - Demo: Creating Firewall ACLs
Configure a DMZ - DMZ's DHCP Address range, Use the DMZ port checklist, CorpDMZWeb
Configure a Perimeter Firewall - From Zone, To Zone, Service, Action, Source Host **Configuring a Proxy Server** - Demo: Configuring a Proxy Server
Firewall Design and Configuration Facts - Configure two firewall devices, demilitarized zone, ARP requests, routed firewall, transparent firewall
Practice Questions - Practice questions

UNIT-IV (LECTURE HOURS: 14)

WAN Concepts
WAN Structure - WAN Structure
WAN Technologies - WAN Technologies
WAN Services - WAN Services
WAN Media Facts - POTS, T1, T3, E1, E3, J3, OC-1
WAN Facts - WAN Cloud, Central Office, Local Loop
WAN Services Facts - PSTN, Long Distance Networks ISDN BRI, Digital Signals over POTS
Practice Questions - Practice questions
WAN Connnections
PPP WAN Connections - Difference between LCP and NCP, which layer does OSI model PPP function, which feature of PPP can detect link errors
Configuring a PPP WAN Link - Configuring a PPP WAN Link
PPP WAN Connection Facts - PPP transmits the password in cleartext, CHAP uses hash, Defines a header and trailer, Multilink PPP
Configure a PPP WAN Link - Configure both serial links to use PPP
Practice Questions - Practice Questions
Traditional Internet Connectivity - What is multiplexing, how does DSL enable you, Connect to a DSL network **Mobile Internet Connectivity** - Mobile Internet Connectivity
Internet Connectivity
Internet Services Facts - PSTN uses a single POTS phone line with a modem, DSL offers digital communications
Connect a DSL Network - SOHO network, RJ11 jacks, DSL router
Practice Questions - Practice questions
Remote Access & WAN Troubleshooting
Remote Access - How does PPPoE differ from PPP; how does RADIUS differ from TACACS+, Proxy ARP **Remote Access Facts** - Physical Connection, Connection Parameters, Authentication
Configuring a Remote Access Server - Demo: Configuring a Remote Access Server
Configuring a RADIUS Solution - Configuring RADIUS Solution
Practice Questions - Practice questions
WAN Troubleshooting - Which utility allows you to verify the application layer connectivity and configuration, up down interface status means
Troubleshooting WAN Issues - Demo: Troubleshooting WAN Issues **WAN Troubleshooting Facts** - Line Status, Protocol Status, Condition **Practice Questions** - Practice questions

UNIT-V (LECTURE HOURS: 14)

Update Management
Update Deployment and Management - Update Deployment and Management
Configuring an Update Server - Demo: Configuring an Update server

SEMESTER - IV
MAJOR-THEORY
BASIC IP SERVICES

Update Deployment and Management Facts - Fix bugs (errors) in programming code, Patch security vulnerabilities, hotfix
Practice Questions - Practice questions
Data Protection
Data Backups - What is the difference between data and server backup , Configure a data backup
Protecting Data - Demo: protecting Data
Recovering Files from Backup - Demo: Recovering Files from backup
Workstation Backup Facts - Types of backup, System image backup, system repair disc
Server Backup Facts - Full server, bare metal recovery, individual volumes, server backup wizard
Configure a Data Backup - Backup all of the users' data files, Include a system image
Configure a Server Backup - Full Server, Backup schedule, Backup locations, CorpWeb
Practice Questions - Practice questions
Remote Management
Remote Management - What is the difference between telnet and SSH, how can remote desktop software differ from terminal emulation software
Using Remote Desktop - Demo: using Remote Desktop
Allow Remote Desktop Connections - Enable Remote Desktop, Networking Closet **Remote Management Facts** - Terminal Emulation, Telnet, SSH, SSH uses TCP port 22 **Practice Questions** - Practice questions
Monitoring
Network Monitoring - Network Monitoring
Protocol Analyzers - Protocol Analyzers
Viewing Event Logs - Demo: Viewing Event Logs
Using a Packet Sniffer - Demo: using a packet sniffer
Monitoring Utilization - Demo: Monitoring Utilization
Monitoring Interface Statistics - Demo: Monitoring Interface Statistics
Network Monitoring Facts - Logs, Pinpoint the source of problems, Load Tester, Throughput Tester
Practice Questions - Practice questions
Log File Management
Log File Management - What does logging process do, what information is provided in the facility component of the log message
Configuring a Syslog Server - Demo: Configuring a Syslog Server
Log File Management Facts - Timestamps, Facility, Severity Level, Mnemonic, Message Text **Practice Questions** - Practice questions
Network Management with SNMP
Network Management - Role of MIB when using SNMP, what is a trap and how can you use it in network administration **Configuring an SNMP System** - Demo: Configuring an SNMP System **SNMP Facts** - Manager, Agent, MIB, Trap

Practice Questions - Practice questions

Text Books :
CCENT/CCNA ICND1 100-105,Official Cert Guide,CCIE NO.1624 | Edition:1 | Ciscopress | Wendell Odom( )

Reference Books :
Computer Networks | Edition:5 | Prentice Hall | Andrew STanenbaum( )

SEMESTER - IV
MAJOR-PRACTICAL
PROGRAMMING LAB- IV (ALGORITHMS & BASIC IP SERVICES)

OBJECTIVES :

- Understand the time complexity of various data structures
- Selecting suitable data structures for any given problem
- Design Optimized algorithm and implement them for a given problem
- To understand the Basic Knowledge in IP Services

HOURS / WEEK - 6		
HOURS / SEMESTER - 90		
CIA	**CO-CURRICULAR ACTIVITIES**	**LECTURE**
12	6	72

UNIT-I (LECTURE HOURS: 18)

Insert an element in Sorted Array

1.Imagine a sorted array named "sarray" of integer type with size 12. Let the elements stored are: 100,99,88,77,66,55,44,33,22,11,10,9. Insert an element : 69. The constraint is that, the array should remain sorted even after the insert operation. Determine the worst case complexity of this process.

Insert an element in Linked List

2. The linked list "LL" has the following elements stored in it. LL=11,22,33,44,55,66. Now try to insert 37 and analyse this operation for the worst case complexity .

Hash Application

3. Create two arrays of same type and store elements in both the arrays. Now try to find, out of these two arrays whether one array is the subset of an another array. State the complexity of your program

UNIT-II (LECTURE HOURS: 18)

Heap

4. Heapify : 856,752,985,659,526 and determine the worst case complexity of your code.

Insertion Sort

5. Assume the given input: 45,85,74,45,12,65,10,6,8,36 and perform the insertion sort and find the worst case complexity of the sorting procedure.

Sequential and Binary Search

6. Implement Sequential search and Binary search and state the complexities of both and reveal what sort is to be used when.

UNIT-III (LECTURE HOURS: 12)

CONFIGURE DHCP

1. Configure DHCP pool in router R2 for Clients connected on its Gigabiethernet 0/1 interface by takingthe following points into account: 1. pool Name :pool1 2. pool Length:192.168.88.0./28 3. Default-Gateway:192.168.88.1 4.DNS-Server Address:4.2.2.2

UNIT-IV (LECTURE HOURS: 12)

CONFIGURE SPAN

2. Configure the SPAN on switch S1's fast ethernet 0/9 interface for bi -directional traffic and monitor the same traffic to the analyzer connected at interface fastethernet 0/22 of the same switch.

UNIT-V (LECTURE HOURS: 12)

CONFIGURE SSH

3. Configure SSH access to Router R1 using following parameters 1. Username: administrator 2. Password: admin@456 3. privilege level: 15 4. Domain: test.com

CONFIGURE OSPF

4. Configure the OSPF routing methods inside the R1, R2 and R3 using following parameters. 1. AREA: Backbone (0) 2. Process id: 1

CONFIGURE SYSLOG SEVER

SEMESTER - IV
MAJOR-PRACTICAL
PROGRAMMING LAB- IV (ALGORITHMS & BASIC IP SERVICES)

5. Configure Syslog server access inside Router R3,R2 to send report to syslog server connected off switch S4 for all events.

Text Books :

Grokking Algorithms | Edition: | Manning | Bhargava AND Aditya Y(2016)

SEMESTER - IV
ALLIED-THEORY
BUSINESS ACCOUNTING

OBJECTIVES :

To get a basic knowledge in accounting principles & various accounting methods used in business.

HOURS / WEEK - 5		
HOURS / SEMESTER - 75		
CIA	**CO-CURRICULAR ACTIVITIES**	**LECTURE**
10	**5**	**60**

UNIT I (LECTURE HOURS: 15)

Accounting

Meaning - Introduction, Meaning Definition
Basic Concepts - Business entity, Going concern, Money measurement, Cost, Dual, Realization, Accounting period, Revenue.
Conventions - Consistency,Disclosure,Conservatism,Materiality
Journals - Meaning, Problems
Ledgers - Meaning, Problems
Subsidiary books - Meaning , Problems
Trial Balance - Meaning, Problems
Final Accounts with simple adjustments. - Meaning, Specimen, Problems

UNIT II (LECTURE HOURS: 10)

Depreciation

Meaning - Meaning
Causes - Wear and tear, Obsolescence, Depletion, time elapses
Need - Ascertainment of profit & loss, cost of production, replacement of assets, keeping capital intact
Methods of Depreciation - Straight line, Diminishing, Annuity, Sinking fund, Insurance policy, Revaluation method
Straight line methods - Meaning, Problems
Written down value method - Meaning, Problems
Change of method - Meaning, Problems

UNIT-III (LECTURE HOURS: 12)

Marginal Costing

Meaning - Meaning
Advantages - Decisions, Better results, Cost control, Apportionment of fixed costs, Realistic
Concepts - Cost volume,profit,variable,fixed.
Break Even Analysis - Cost-volume-profit, total revenue=total cost, No profit no loss
Uses and assumptions. - Fixed and variable costs, constant at all levels of ouput, vary with the volume of output, no change in operating efficiency.
Applications of Maginal costing - Problems

UNIT IV (LECTURE HOURS: 13)

Budget and Budgetary Control

Meaning - Meaning
Definition - Definition
Objectives - Define goal of the enterprise, long and short term plans, co-ordinate the activities, eliminate waste, centralize control system, correct deviations.
Advantages - Objectives & policies, controlling, co-ordination, fix up responsibility, standard costing. **Limitations** - Government policies, decision making, heavy expenditure, active & passive resistance, co-operation

SEMESTER - IV
ALLIED-THEORY
BUSINESS ACCOUNTING

Steps in Budgetary Control System - Establish budget, actual performance, continuous comparison, suitable remedial action, revision.
Classification of Buget. - Sales,production,purchase,cash,flexible,master,zero based budget, problems

UNIT V (LECTURE HOURS: 10)

Accounting Package

Tally Introduction - Software package, power, payroll, inventory.
Company Info - Create company, creation.
Gateway of tally - Accounts info, inventory, voucher, stock.
Masters - outsource
Transactions / Vouchers - outsources
Reports, general ledger - Sales control, analysis, stock analysis, cash & credit, security control
Balance Sheet, Profit & Loss a/c,Stock Summary - outsources

Text Books :

T1.Business Accounting | Edition:11 Edition | Teachers publishing house | DR.JAYACHANDRAN.(2008)
T2.Management accounting | Edition:12 Edition | Sriram Publications-2011 | DR. R. SRINIVASAN.(2011)
T3.Tally | Edition:7 Edition | Nels publication | NELLAI KANNAN.(2008)

Reference Books :

R1.Double Entry book keeping | Edition:5 Edition | Sultan Chand & Sons | T.S.GREWAL.(2008)
R2.Cost Accounting | Edition:6 Edition | Kalyani Publisher | K.L.NARANG. AND S.P.JAIN.(2007)

SEMESTER - IV
ELECTIVE-THEORY
ELECTIVE-I WEB BASICS

OBJECTIVES :

- To study fundamental concepts and terms that is need for using HTML and CSS .
- To show how to enter, edit, test and validate HTML and CSS for the web pages of a website .
- To define the contents and structure for a web page and shows how to code the CSS that does basic formatting to the HTML content.
- To learn how to use the CSS box model for spacing, borders, and backgrounds layout and to use CSS for page as well as to work with links and lists.

HOURS / WEEK - 5		
HOURS / SEMESTER - 75		
CIA	**CO-CURRICULAR ACTIVITIES**	**LECTURE**
10	**5**	**60**

UNIT I (LECTURE HOURS: 12)

An Introduction to HTML and CSS

The HTML and CSS for a web page - HTML document, DOCTYPE declaration

A short history of the HTML and CSS standards. - HTML elements, style sheet, rule set

THE HTML SYNTAX

The basic structure of an HTML document - HTML elements, root element

How to code elements and tags - two elements with opening and closing tags, two empty tags, correct & incorrect nesting of tags

How to code Attributes - opening tag with attributes, empty tag with attribute, Boolean attribute, coding rules

How to code comments and whitespace - An html Document with comments and white space

The CSS syntax

How to code CSS rule sets and comments - Rule set, selector, declaration, comments.

How to code basic selectors - Element type, id or class.

HOW TO USE APTANA TO WORK WITH HTML AND CSS FILES

How to create a project - Creating a project

How to open an HTML file - opening an HTML file within a project

How to start a new HTML file - start a new file from any template

How to edit an HTML file - Aptana with an auto completion list for an HTML file common coding errors

How to open or start a CSS file - Start a new CSS file for a project from any template

How to edit a CSS file - Aptana with an auto-completion list for a CSS file, common coding errors

How to preview and run an HTML file - show preview and run buttons

UNIT II (LECTURE HOURS: 13)

HOW TO CODE THE HEAD SECTION

How to code the title element - Speakers and luncheons

How to link to a favicon - The link element is used to link a custom icon **How to include metadata** - a head section includes metadata, SEO guidelines

HOW TO CODE TEXT ELEMENTS

How to code headings and paragraphs - Block elements h1 to h6, p

special blocks of text - pre , blockquote , address

How to code inline elements for formatting and identifying text - Formatting text such as i, b, sub, sup, br and identifying content are abbr, cite, code etc.,

Character and core attributes. - character entities are & amp, & lt, & gt etc. attributes are id, class, title, lang.

How to code links, list, and images

How to code absolute and relative URLs - Root- relative path, document-relative path

How to code links and lists - href attribute for < a > element, link to a web page in same folder, link subfolder of the parent folder, link based root folder, link to another website, Elements created ordered and unordered lists < ul >, < ol >, < li >

How to include images - Attributes of < img > element are src, alt, height, width.

An Introduction to CSS

Three ways to provide CSS styles for a web page - External style sheet, embedded style sheet, inline styles

SEMESTER - IV
ELECTIVE-THEORY
ELECTIVE-I WEB BASICS

How to specify measurements and colors

How to specify measurements - Common units of measure such as px, pt, em, %, relative units of measure with a fixed border.

How to specify colors - specify colors, with a RGB value, with a RGB value that uses hexadecimal numbers

HOW TO WORK WITH CASCADING STYLE SHEETS

How the cascade rules work - User rule set, identify rule as important, cascade order for applying CSS rule set.

How to use the developer tools to inspect the styles that have been applied - Modern browser provide developer tool to inspect elements and styles.

HOW TO WORK WITH TEXT

How to set the font family and font size - Generic font families, a font-family rule in the body element that is inherited by all descendants, a font-family rule in a descendant that overrides the inherited font family.

How to set the other properties for styling fonts - syntax for the shorthand font property.

How to set properties for formatting text - Properties for indenting, aligning & decorating text, CSS that specifies a text indent & horizontal alignment.

How to float an image so text flows around it - Float property is used to float an image & set margin around it using margin property.

UNIT III (LECTURE HOURS: 12)

An introduction to the box model

How the box model works - Formula for calculating height & width of a box.

A web page that illustrates the box model - CSS adds for types of border to the elements in HTML

HOW TO SIZE AND SPACE ELEMENTS

How to set heights and widths - Properties for setting height & width, set the height & width of the content area, set maximum & minimum width & height.

How to set margins and padding - Margins - margin-top, margin-right, margin-bottom, margin-left. Padding - padding-top, padding-right, padding-bottom, padding-left.

How to set borders and backgrounds

How to set borders - Border - Borders, syntax, set side border, width of border, style & color. **How to set background colors and images** - Properties for setting background color & image

HOW TO CODE LISTS

How to code unordered and ordered lists - Unordered list is displayed as bulleted list, list items can contain inline elements, ordered list is displayed as numbered list.

How to code nested lists and description lists - Nest lists by coding one list as an item for another list, description list, definition list.

HOW TO FORMAT LISTS

How to change the bullets for an unordered list - Properties for formatting unordered list, values for list-style-type property.

How to change the numbering system for an ordered list - Values for the list-style-type property of an ordered list.

How to change the alignment of list items - Margin & padding to control the indenting for the item in an ordered or unordered list.

HOW TO CODE LINKS

How to link to another page - Four attributes of the < a > element

How to format links - Common CSS pseudo-classes for formatting links, properties for removing underlines & borders.

How to use a link to open a new browser window or tab - Target attribute of the < a > element is set to open a new window.

How to create and link to placeholders - Placeholder or anchor use id attributes of the < a > tag.

How to link to a media file - Media formats & MIME types, plugins.

How to create email, phone, and Skype links - Link type: E-mail, prefix mailto:, phone: tel:, skype: skype prefix is used.

UNIT IV (LECTURE HOURS: 11)

Basic skills for working with images

Types of images for the Web - Jpeg, gif, png & description

How to include an image on a page and to resize an image - Attributed of the < img > tag, CSS properties for sizing an image.

How to align an image vertically and to float an image - Property for aligning images vertically: vertical-align, common keywords. Property for floating images: float, clear.

ADVANCED SKILLS FOR WORKING WITH IMAGES

How to work with thumbnails - To download an image faster & can use image editor for sizing an image.

How to do image rollovers - Image rollover to use background images

SEMESTER - IV
ELECTIVE-THEORY
ELECTIVE-I WEB BASICS

How to create image maps - image map attributes are usemap, name.

Basic HTML skills for coding tables

An introduction to tables - Rows, columns, data cells, header, footer.

How to create a table - Elements of coding table: table, tr, th, td.

How to add a header and footer - Thead, tbody & tfoot elements make easier to style a table with CSS.

Basic CSS skills for formatting tables

How to use CSS properties to format a table - Formatting table - Border-collapse, border-spacing, padding, text-align, vertical-align, tr, td & th elements

OTHER SKILLS FOR WORKING WITH TABLES

How to merge cells in a column or row - Merge cells - Colspan & rowspan

How to provide for accessibility - Accessibility - Caption, headers & scope.

How to nest tables and control wrapping - A table with another table nested within one of its cells, Content of the cells in a table - wrap.

UNIT V (LECTURE HOURS: 12)

HOW TO USE FORMS AND CONTROLS

How to create a form - Form - name, action, method, and target. Controls, HTTP request, get method.

How to use buttons - Button - Type, value, src, alt, height & width.

How to use text fields - Text field - Text field, password field, hidden field. Type, value, maxlength, size, autofocus, placeholder

How to use radio buttons and check boxes - Radio button & check boxes - Type, value, checked

How to use drop-down lists and list boxes - The optgroup & option elements - Label, value, selected. List boxes - Size, multiple.

How to use text areas and labels - Textarea element - Rows, cols, wrap. Label element - for.

How to group controls with fieldset and legend elements and to use a file upload control - Group controls, legend element coded within a field element, disabled attribute. File upload control - Accept, multiple

Other skills for working with forms

How to align and format controls - Series of controls and labels are align, text-align property, rule set coded for input control, :focus pseudo-class to change appearance of a control.

How to set the tab order and assign access keys - Tabindex, accesskey attributes for setting the tab order & access keys.

AN INTRODUCTION TO MEDIA ON THE WEB

Common media types for video and audio - Common media types for video & audio, media player, plugins.

Video codecs - H.264, Theora, VP8

Audio codecs - AAC, FLAC, MP3, Vorbis, WMA.

Audio and video support in current browsers - Audio & video types, mobile devices, MIME type.

How to encode media - Encoder - Miro Video Converter.

HOW TO ADD AUDIO AND VIDEO TO A WEB PAGE

How to use the object and param elements - Object - type, data, width, height, param - name, value.

How to use the embed element - Embed element - type, src, width, height.

How to fall back to Flash for backward compatibility - HTML for playing a video & audio that falls back to Flash

Text Books :

T1: Murach's HTML5 and CSS3 | Edition:3rd Edition | SHROFF PUBLICATION & DISTRIBUTERS PVT.LTD. | Anne Boehm AND Zak Ruvalcaba(2016)

Reference Books :

T2 : Brilliant HTML5 and CSS3 | Edition:2011 | Pearson | James A. Brannan AND Josh Hill( )

ELECTIVE-I

COMPILER DESIGN

Total Lecture Hours: 60

Objective:

- ✓ To understand design and implement a lexical analyzer

Pedagogy:

Lecturing, Group Discussion, LCD, OHP and Seminar

UNIT – I **(8 HOURS)**

Compilers – Analysis of the source program – Phases of a compiler – cousins of the compiler – grouping of phases – compiler construction tools – lexical analysis – role of lexical analyzer – input buffering – specification of tokens.

UNIT – II **(10 HOURS)**

Syntax Analysis: Role of the parser - Writing Grammars – context – free grammars – top down parsing – recursive descent parsing – predictive parsing – bottom-up parsing – shift reduce parsing – operator precedent parsing.

UNIT – III **(10 HOURS)**

Intermediate Code Generation: Intermediate languages– Assignment statements – Boolean expressions case statements – back patching- procedure calls.

UNIT – IV **(10 HOURS)**

Code Generation: Issues in the design of code generator - The target machine – runtime storage management – basic blocks and flow graphs – next-use information – a simple code generator –peephole optimization.

UNIT – V **(10 HOURS)**

Code Optimization: Introduction – principal sources of optimization – optimization of basic blocks – introduction to global data flow analysis – runtime environments – source language issues – storage organization – storage allocation strategies – access to non-local names – parameter passing.

TEXT BOOKS:

1. S Sudha, Principles of Compiler Design , Sams Publishers ,2010
2. Alfred Aho, Ravi Sethi, Jeffrey, Ullman, Compilers Principles, Techniques and tools, Pearson education Asia , 2003.

SEMESTER - IV
SKILL BASED COURSE-THEORY
APTITUDE SKILLS

OBJECTIVES :

A humble attempt in helping the students to score a good percentage in the Aptitude Test to overcome the written test in interviews successfully

HOURS / WEEK - 3		
HOURS / SEMESTER - 45		
CIA	**CO-CURRICULAR ACTIVITIES**	**LECTURE**
6	**3**	**36**

UNIT I (LECTURE HOURS: 8)

H.C.F & L.C.M
H.C.F & L.C.M - Related Problems
Number Simplification
Number Simplification - Related Problems
Ratio
Ratio - Related Problems
Average
Average - Related Problems

UNIT II (LECTURE HOURS: 7)

Percentage
Percentage - Related Problems
Simple and Compound Interest
Simple and Compound Interest - Related Problems
Profit and Loss
Profit and Loss - Related Problems

UNIT III (LECTURE HOURS: 7)

Problems on Ages
Problems on Ages - Related Problems
Time and Distance
Time and Distance - Related Problems
Problems on Trains
Problems on Trains - Related Problems

UNIT IV (LECTURE HOURS: 7)

Time and Work
Time and Work - Related Problems
Probability
Probability - Related Problems

UNIT V (LECTURE HOURS: 7)

Permutations and Combinations
Permutations and Combinations - Related Problems
Allegation and Mixtures
Allegation and Mixtures - Related Problems

Text Books :

QUANTITATIVE APTITUDE | Edition: | S.CHAND & COMPANY PVT.LTD,NEW DELHI | Dr.R.S.AGGARWAL(1989)

SEMESTER - IV
SKILL BASED COURSE-THEORY
APTITUDE SKILLS

Reference Books :

TEST OF REASONING & QUANTITATIVE APTITUDE | Edition: | S.CHAND & COMPANY PVT.LTD,NEW DELHI | P.K.AGARWAL(2004)

SEMESTER - V
MAJOR-THEORY
RELATIONAL DATABASE MANAGEMENT SYSTEM

OBJECTIVES :

1. To enable the students to understand the concepts of Database Management System
2. To understand the Normalization process and ER modeling
3. To create various queries to solve complex scenarios

HOURS / WEEK - 5		
HOURS / SEMESTER - 75		
CIA	**CO-CURRICULAR ACTIVITIES**	**LECTURE**
10	**5**	**60**

UNIT I (LECTURE HOURS: 10)

Introduction to database concepts

Introduction - Basic building block- collection of facts, Qualitative Vs Quantitative, Example: What do we know about the Dog?
Database - Definition
Database Management System - Definition and purpose
Database Applications - Enterprise , banking and Finance, characteristics of a good DB
File processing systems - Characteristics of a good database, Data Storage relationship, Isolated, Example
Advantages over file systems - Reduces Data redundancy and data inconsistency, Data isolation, Data integrity, Atomicity of operations, Concurrency , Security
Data Abstraction - View of Data, University Database Example

RELATIONAL DATA MODEL

Alternative Terminology
Introduction - Attributes, Domain, Example
Database Schema and Instance - Definition Relational Schema, Relational instance Example
Physical Data Independence - Definition, Example

KEYS

Candidate key
Introduction - Purpose of the keys
Super Key - Definition, Example
Foreign key
Primary key - Points to remember for a Primary key
Secondary Key - Definition, Example
Query Languages - Procedural language Nonprocedural language Definition, Example

Tutorial Hour

Keys - Defining Various key from a database.

UNIT II (LECTURE HOURS: 11)

BASIC STRUCTURE SQL

Introduction - Declarative language Example
Several types of SQL - DDL, DML, Integrity, View, Transaction, Embedded SQL, Authorization
Select clause - Syntax and Examples
From clause - Syntax and Examples
Where clause - Syntax and Examples

Creating and
Renaming a relation

Create table - Syntax and example
Tuple Variables
The Rename Operation - Syntax and example

String Operations

SEMESTER - V
MAJOR-THEORY
RELATIONAL DATABASE MANAGEMENT SYSTEM

Like - Percent (%), Underscore (_)
Display of Tuples
order by clause - Ascending and descending order
Set operations - Union Operation ,Union all, Intersect Operation, Except Operation
Tutorial Hour
Table Definition - Basic of table creation and various clauses

UNIT III (LECTURE HOURS: 15)

Aggregate and Grouping Functions
Basic Aggregation - Average, Sum ,Count, Maximum, Minimum
group by - Syntax and example
Having - Syntax and example
Modification of the Database
Insertion - Syntax and example
Deletion - Syntax and example
Updates - Syntax and example
Data types - Standard and Other
JOIN EXPRESSIONS
natural join operation
Join types and Join conditions - Syntax and example
Outer join - Left right, full outer join
View Definition - Syntax and example
CONSTRAINTS
Unique Constraint
The check Clause
Default Values
Not null constraint - Syntax and Example
alter table
Primary key constraint - Syntax and Example
Referential Integrity - Add, Modify, drop table
Tutorial Hour
Sql operations - Querying

UNIT IV (LECTURE HOURS: 14)

Relational Database Design
Features of Good Relational Designs - Scenarios and Examples
FUNCTIONAL DEPENDENCY (FD)
Introduction - FDs are domain knowledge - DB engine will not help, no optimization, Relation satisfying a dependency Vs Dependency holding on a schema
Types of FD - Trivial, Non trivial, Completely non-trivial
Minimal FDs - Definition, Example
Closure and covers of set of functional dependencies
Dependency Preservation - Definition, Example
Armstrong's axioms - Reflexive, Augmentation, Transitive, Decomposition, Union, Pseudotransitivity
Basic Normal forms
First Normal Form (1NF) - 1NF: based on attributes only
Second Normal Form (2NF) - Prime attribute, full functional dependency and partial dependency
Third Normal Form (3FN) - Transitive Dependency
Higher Normal Forms
Boyce-Codd normal (BCNF) - Normal forms, Lossless decomposition, Anomalies with BCNF **Fourth normal form (4NF)** - Multi-valued dependency (MVD), MVD and lossless join
Fifth normal form (5NF) or Project-Join normal form (PJNF) - Join dependency (JD)
Domain-Key normal form (DKNF) - Syntax and Example
Tutorial Hour
Normalization - Complete Example

UNIT V (LECTURE HOURS: 10)

THE ER MODEL CONSTRAINTS

SEMESTER - V
MAJOR-THEORY
RELATIONAL DATABASE MANAGEMENT SYSTEM

Relationship Sets
Attributes
Entity Sets - Definition and example
Participation Constraints
Keys
What to remove
Mapping Cardinalities - Definition and example
ENTITY-RELATIONSHIP DIAGRAMS
Mapping Cardinality
Complex Attributes
Basic Structure - Example
Weak Entity Sets E-R diagram for the University Enterprise
Roles - Example
Non binary Relationship Sets - Example - diagram
REDUCTION TO RELATIONAL SCHEMAS
Representation of Strong Entity Sets with Complex Attributes
Representation of Strong Entity Sets with Simple Attributes - Example
Representation of Weak Entity Sets - Example
Representation of Relationship Sets - Redundancy of Schemas, Combination of Schemas
EXTENDED E-R FEATURES
Generalization
Inheritance
Specialization - Example
Tutorial Hour
ER Diagram Creation - Example

Text Books :
Database System Concepts | Edition:6th | TMH Publications | Korth AND Silberschatz AND Sudarshan(2011)

Reference Books :
Beginning Database Solutions | Edition: | Wrox Publications | Rod Stephens(2009)
Database System Concepts | Edition:4th | TMH Publications | Silberschatz-Korth-Sudarshan(2001)
Fundamentals of Database Systems | Edition:6th | Pearson Education publication | , Shamkant B. Navathe AND Ramez Elmasri(2011)

SEMESTER - V

MAJOR-PRACTICAL

PROGRAMMING LAB-V(RELATIONAL DATABASE MANAGEMENT SYSTEM AND PHP & MY SQL)

OBJECTIVES :

HOURS / WEEK - 6		
HOURS / SEMESTER - 90		
CIA	CO-CURRICULAR ACTIVITIES	LECTURE
12	6	72

UNIT I (LECTURE HOURS: 12)

1. Display greeting message based on IST timezone

Design a PHP program to greet the users, who visit the homepage of a particular website according to current timein PHP. The date function with parameter 'H' is used to find Current Time. This program works according to ISTtime zone. (a)If the user enters the time less than 12, it displays a greeting message as "Good Morning" followed byrelevant quotes and greeting images. (b)If the user enters the time between 12 to 17, it displays a greeting messageas "Good Afternoon" followed by relevant quotes and greeting images. (c)If the user enters the time greater than 18,it displays a greeting message as "Good Night", followed by relevant quotes and greeting images.HINT: Use date/timefunction to fetch the instant time from the server where your PHP script runs.

2. Product Ranking System

Build a web page that performs a product rating, of latest mobile phones available in the market. The product ratingis carried out using the grade scale like good, very good, excellent in a web form. The user must select any one ofthe grading scale according to the various features available in each product like sound, battery, video, display,cost, etc. The user can also provide their own suggestions to improve branding.HINT: $_Post is a variable used to fetchthe form elements values.

UNIT II (LECTURE HOURS: 12)

3. Multi-dimensional Array

Rewrite the large cities array into a multi-dimensional array called $multiCity. The first sub-array will be completelynew and include the labels, City, Country, Continent. Each of the succeeding sub-arrays should include those threeitems, one for each of the cities, for a total of 6 sub-arrays.Here's the content for your array: City, Country,Continent; Tokyo, Japan, Asia; New York City, USA, North America; Mumbai, India, Asia; Shanghai, China, Asia;London, UK, Europe.In the HTML, use the array in a table. The first row should be a header row and contain theentries in the first sub-array. Call these items without using a loop. For the succeeding rows, use a for loop with aforeach loop nested inside to populate the table with the remaining contents of the array. Use the count() functionso that your for loop will function properly even if you increment or decrement the array.HINT: Add a simple style sheet inthe head section of your HTML.

4.Pragathi store database manipulations

Create a database_table named pragathi_store and make various entries like item _number, item_name, quantity,unit_cost, total _purchase_cost. Use PhpMyAdmin to insert more than 5 records.Performupdation and deletion ofrecords based on relevant constraints.HINT: 'Edit item_name based on item_number for a particular record' is one of theexample constraints that can be setforth in query.

UNIT III (LECTURE HOURS: 12)

5.Display Customer Preferences from MySQL in Table Format

SEMESTER - V

MAJOR-PRACTICAL

PROGRAMMING LAB-V(RELATIONAL DATABASE MANAGEMENT SYSTEM AND PHP & MY SQL)

Code a PHP program that fetch the customer information like Cust_no, Cust_name, Item_purchased and Mob_nofrom the MySQL database table. Use mysql_connect() to connect from PHP to MySQL and mysql_query() to fetch allthe records from a database table named "customer". Before starting with PHP code create a table with all therelevant fields in PHPMyAdmin.HINT: Display all the information about a customer in table format on the web page.

6.Feedback system

Create a simple webpage about cosmetic products by inserting images, video files and a feedback form, fetchvaluable suggestions from the customer end. Create a table named cosmetics with various fields like name, age,gender, phone no and feedback before running the source code of PHP. HINT:Use .css to create a stylesheet and storethe feedbacks into database table using MySQL.

UNIT IV (LECTURE HOURS: 36)

Problem Domain

• The bank is organized into branches. Each branch is located in a particular city and is identified by a unique name. The bank monitors the assets of each branch.

• Bank customers are identified by their customer-id values. The bank stores each customer's name, and the street and city where the customer lives. Customers may have accounts and can take out loans. A customer may be associated with a particular banker, who may act as a loan officer or personal banker for that customer.

• Bank employees are identified by their employee-id values. The bank administration stores the name and telephone number of each employee, the names of the employee's dependents, and the employee-id number of the employee's manager. The bank also keeps track of the employee's start date and, thus, length of employment.

• The bank offers two types of accounts-savings and checking accounts. Accounts can be held by more than one customer, and a customer can have more than one account. Each account is assigned a unique account number. The bank maintains a record of each account's balance, and the most recent date on which the account was accessed by each customer holding the account. In addition, each savings account has an interest rate, and overdrafts are recorded for each checking account.

• A loan originates at a particular branch and can be held by one or more customers. A loan is identified by a unique loan number. For each loan, the bank keeps track of the loan amount and the loan payments. Although a loan payment number does not uniquely identify a particular payment among those for all the bank's loans, a payment number does identify a particular payment for a specific loan. The date and amount are recorded for each payment.

Creating and renaming a relation - Create Table, As clause

Constraints - Not null, Primary Key, Foreign Key, Unique Key, Default, Check

Modification of the Database - Insert, Delete, Update

String Operations - Select operation, Like Clause

Display of Tuples - Select operations

Aggregate and Grouping Functions - Sum, Count, Avg, Min , Max, Group By, Having

Join Expressions - Natural join, Inner join, Outer join

Sub Queries - Nested Queries

Text Books :

Database System Concepts | Edition:6th | TMH Publications | Korth AND Silberschatz AND Sudarshan(2011)

SEMESTER - V
COMMUNICATIVE ENGLISH-THEORY
CAREER SKILLS III

OBJECTIVES :

i. To create an awareness on the importance of Soft Skills and to help the students acquire and refine them .
ii. To make the students industry ready by enhancing their communication and soft skills .

HOURS / WEEK - 1		
HOURS / SEMESTER - 15		
CIA	**CO-CURRICULAR ACTIVITIES**	**LECTURE**
2	**1**	**12**

UNIT 1 (LECTURE HOURS: 3)

Grooming & Personality Development

Respect, morals, values about the system & behavior - Ethics to be followed inside the institution and in an organization during their future career advancement.

Speak-up

JAM - Speech on any given topic. It gives a chance for the students to identify themselves and build their confidence.

UNIT 2 (LECTURE HOURS: 2)

Corporate Etiquette

Dressing & Basic Etiquettes - How to behave socially in a workplace, punctuality, grooming and interpersonal communication within the organization.

Team Building

Steps in building team and skills required - How to build a team, How to be a team player and team leader

UNIT 3 (LECTURE HOURS: 2)

3 Min Presentation

Presentation on any given topic - Presentation, Evaluation & Feedback

UNIT 4 (LECTURE HOURS: 3)

SWOT Analysis

Analysing oneself - Strength, Weakness, Opportunity and Threat to achieve a certain goal

Self Introduction

Practice - Correction, Evaluation & Feedback

UNIT 5 (LECTURE HOURS: 2)

Group Discussion

Practice - Correction, Evaluation & Feedback

Text Books :

Soft Skills | Edition:3 | Bloomsbury | Dr. HyacinthPink(2015)

Reference Books :

Personality Development & Soft Skills | Edition:2 | Oxford University Press | Barun KMitra(2016)

SEMESTER - V
MAJOR-THEORY
PHP & MY SQL

OBJECTIVES :

OBJECTIVES:

· Understand and demonstrate analytical powers of thought through critical analysis, evaluation and synthesis of fundamentals of PHP language and its associated Technologies .

· Demonstrate awareness on using PHP to create dynamic interactive Web forms .

·To create flexibility in designing, planning and building real-world projects.

·Develop specific focus and depth of critical understanding and interpretation in specific areas in the Web Development

HOURS / WEEK - 6		
HOURS / SEMESTER - 90		
CIA	**CO-CURRICULAR ACTIVITIES**	**LECTURE**
12	6	72

UNIT I (LECTURE HOURS: 15)

WHAT IS PHP?
&
WHY USE PHP ?

A dynamic web page - Interactive web site

Process of running PHP script - Clicking a link, requested URL, Script runs

Common examples of PHP scripts - Web forum, Online stores, Search enginees,Blogs,Web mail applications **Feature of PHP** - Cross Platform, Integrate with all web servers

How does PHP compare with other common Web programming technologies? - Asp,Asp.Net,Perl,Java,Python,Ruby, ColdFusion

120THE EVOLUTION OF PHP
&
WHAT'S NEW IN
PHP 5.3

Early history of PHP - PHP version 2,PHP version 3.3,PHP 4,PHP 5

Namespaces - Avoid naming clashes

The goto Operator - Example

Nowdoc Syntax - Embed a block of PHP code

Shorthand Form of the Ternary Operator - Syntax

Advanced Changes - Late static bindings, Optional garbage collector

INSTALLING PHP

Installing on Ubuntu Linux - Steps to install on Ubuntu Linux

Installing on Windows - Steps to install WampServer

Installing on Mac OSX - Steps to install on Mac OS

Testing Your Installation - Testing the Web Server

Testing PHP - Open text editor,type port number,phpinfo()

Setting Your Time Zone - Steps to set your time zone

OTHER WAYS TO RUN PHP

Running PHP with other Web Servers - Internet Information Server, ISAPI module

Compiling PHP Yourself - Basic steps to compile PHP

Running PHP Remotely - Ssh package

CREATING YOUR FIRST SCRIPT

Embedding PHP within HTML - Code to create a Stylish Page, embed PHP within an HTML page

Enhancing the Script Further - Example to enhance the script to display the current time

Using Comments to Make Code More Readable - Single line comments and multi - line comments

USING VARIABLES IN PHP & UNDERSTANDING DATA TYPES

Naming Variables - Rules for naming variables, example

Creating Variables - Initializing a variable, example of declaring and initializing a variable

About Loose Typing - Loosely - typed language,example

Testing the Type of a Variable - gettype() function, PHP ' s type testing functions

SEMESTER - V
MAJOR-THEORY
PHP & MY SQL

Changing a Variable's Data Type - settype() function,example
Changing Type by Casting - List of casting functions that used in PHP
OPERATORS AND EXPRESSIONS
Operator Types - Arithmetic, assignment, logical, comparison, error control, incrementing-decrementing, Array, string
Understanding Operator Precedence - Precedence of Some PHP Operators (Highest First) **Constant values** - define() function
MAKING DECISIONS
Simple Decisions with the if statement - if syntax and its example
Providing an Alternative Choice with the else statement - if…els, else…if syntax and its example
Testing one Expression Many Times with the switch statement - switch statement- syntax and its example
Compact coding with the Ternary Operator - ternary operator syntax with example
180DOING REPETITIVE TASKS WITH LOOPING
Simple Looping with the while statement - while syntax with example
Testing at the End: The do…while Loop - do…while syntax with example
Neater Looping with the for statement - The general syntax of a for loop
Escaping from Loops with the break statement - Explanation on break statement with an example
Skipping Loop iterations with the continue statement - Explanation on continue statement with an example
Creating Nested Loops - A Homing Pigeon Simulator:Application
Mixing Decisions and Looping with HTML - Fibonacci sequence example

UNIT II (LECTURE HOURS: 15)

CREATING AND ACCESSING STRINGS
Including More Complex Expressions within Strings - Common escape sequences
Using Your Own Delimiters - heredoc syntax and nowdoc syntax
Other ways to create Strings - PHP function return string values
Finding the Length of a String - strlen() , str_word_count() functions
Accessing Characters within a String - About substr() function
SEARCHING AND REPLACING TEXT WITHIN STRINGS
Searching Strings with strstr() - Syntax and usage of strstr() function with example
Locating Text with strops() and strrpos() - Syntax and usage of strops() and strrpos() function with example
Finding the Number of Occurrences with substr_count() - Syntax and usage of substr_count() function with example **Searching for a Set of characters with strpbrk()** - Syntax and usage of strpbrk()function with example
Replacing All Occurrences using str_replace() - Syntax for str_replace(),substr_replace() and strstr() syntax with example
Replacing a Portion of a String with substr_replace() - Syntax for str_replace(),substr_replace() and strstr() syntax with example
Translating Characters with strstr() - Syntax for str_replace(),substr_replace() and strstr() syntax with example
DEALING WITH UPPER- AND LOWERCASE:
FORMATTING STRINGS
String functions for converting upper and lower cases - Syntax for strtolower(), strtoupper(), ucfirst(), lcfirst() functions **General- Purpose Formatting with printf() and sprint()** - printf() versus sprintf(),Using type specifiers, Specifying Signs, Padding the Output, Specifying Number Precisions, Swapping arguments, Storing the Result Instead of Printing it
Trimming Strings with trim(),ltrim() and rtrim() - Purpose of trim functions and its syntax **Padding Strings with str_pad()** - Adding space to the left and right of the string
Wrapping Lines of Text with wordwrap() - Splitting of lines to several lines using wordwrap() **Formatting Numbers with number_format()** - Purpose of number_format function and its syntax
CREATING & ACCESSING ARRAY ELEMENTS
Two types of array - Indexed arrays , Associative arrays,how does PHP arrays work
Creating a new array variable - built-in array() construct **Changing Elements** - Example
Outputting an Entire Array with print_r() - Explanation on print_r()
Extracting a Range of Elements with array_slice() - Explanation on array_slice()
Counting Elements in an Array - Explanation on count()
Stepping Through an Array - Manipulating pointer using current(), key(), next(), prev(), end() and reset() function and its descriptions
LOOPING THROUGH ARRAYS WITH FOREACH
Using foreach to Loop Through values - Syntax for - foreach… loop and a demo program to retrieve key and value pairs
Using foreach to Loop Through Keys and Values - Syntax for - foreach… loop and a demo program to retrieve key and value pairs **Altering Array Values with foreach** - Syntax for - foreach… loop and a demo program to retrieve key and value pairs

SEMESTER - V
MAJOR-THEORY
PHP & MY SQL

WORKING WITH MULTIDIMENSIONAL ARRAYS

Creating a Multi- dimenstional Array - Demonstration of displaying an array of Books application

Accessing Elements of Multidimensional Arrays - Demonstration of displaying an array of Books application

Looping through Multidimensional Arrays - Demonstration of displaying an array of Books application

WHAT IS A FUNCTION? & WHY FUNCTIONS ARE USEFUL

Definition - Subroutine,calling code

Reasons to use functions - Avoid duplicating code, easy to eliminate errors,reuseablitity

Calling
&
Working with Variable Functions

A function call - Syntax for calling function and example

WRITING YOUR OWN FUNCTIONS

Defining Parameters - Use of function with defined parameters and default values

Optional Parameters and Default values - Use of function with defined parameters and default values

Returning Values from your Functions - Understanding local and global variable

Understanding Variable Scope - Understanding local and global variable.

Writing Recursive Functions - Definition,base case,recursive case,overview of recursive function, Creating the Fibonacci Sequence with Recursion

UNIT III (LECTURE HOURS: 14)

How HTMLForms Work

Form tag with its attributes - Action,method,example, How it works, form fields, form elements with get and post method

Capturing Form Data with PHP

Superglobal arrays - $_GET, $_POST, $_REQUEST

Dealing with Multi-Value Fields
&
Generating Web Forms with PHP

Adding multi-value fields in php script - adding square brackets ([]),example, two common approaches to generate a form within PHP

Saving State with Query Strings

Building & Accessing Data Query Strings - Query string,query string characters to be used within field names and values, http_build_ query() function, $_GET superglobal array

WORKING WITH COOKIES

Cookie Components - What is cookies,example to create a cookie,fields in cookie **Setting a Cookie in PHP** - setcookie()function,example to use setcookie()

Accessing & Removing Cookies in Your Scripts - $_COOKIE superglobal array,example

USING PHP SESSIONS TO STORE DATA

Creating a Session - session_ start() function

Reading and Writing Session Data - $_SESSION[]

Destroying a Session - session_destroy() function, $_SESSION array, session_name()

Passing Session IDs in Query Strings - PHPSESSID field, session_id() function

Changing Session Behavior - Directives in php.ini file

Understanding Files and Directories

File & Directories - Importance of files and directories

OPENING AND CLOSING FILES

Opening a File with fopen() - fopen()function usage and explanation

Closing a File with fclose() - fclose() function usage and explanation

Reading and Writing to Files

Functions to read and write the files - About fread(),fwrite(),fseek(),feof(),fgets(), fgetc()

Working with Dates and Times

Getting the Current date and time - date(), getdate(), setdate(), checkdate(), time(), mktime()

CREATING AND MANIPULATING IMAGES

Creating a New image - About imagecreate()

Allocating colors - imagecolorallocate(), imagecolorresolve(),imagejpeg(), imagegif(),imagepng(),imagesetpixel(),imageline(),imageline(), imageellipse(),imagerectangle(), imagepolygon(), imagearc() **Outputting Images** - imagecreatefromjpeg(),imagecreatefromgif(),imagecreatefrompng()

SEMESTER - V
MAJOR-THEORY
PHP & MY SQL

Drawing in an image - imagecreatefromjpeg(),imagecreatefromgif(),imagecreatefrompng()
Opening an Existing Image - imagecreatefromjpeg(),imagecreatefromgif(),imagecreatefrompng()

UNIT IV (LECTURE HOURS: 14)

HOW TO HANDLE EXCEPTIONS

How to create and throw exceptions - The syntax for a try-catch statement, A try-catch statement that catches an exception object, A try-catch statement that throws an exception object, A try-catch statement that catches two types of exception ,description

How to use the try-catch statement - The syntax for creating a new exceptions, The syntax for the throw an exception, A statement that causes an exception to be thrown, methods of exception objects

SETTING UP MYSQL

Starting the MySQL Server - The MySQL server and its command-line tool, Ubuntu-WAMP Server on Windows, MAMP on Mac OS X

Setting Up the MySQL root Password - About root directory, root password, setting privileges

A QUICK PLAY WITH MYSQL

Creating a Table
Adding Data to a Table
Reading Data from a Table
Updating Data in a Table
Deleting Data from a Table
Deleting Tables and Databases

Creating a New Database - Explaination on CREATE DATABASE command,use command,create table,show tables, Insert,Update,Delete,Drop Commands and its examples

CONNECTING TO MYSQL FROM PHP

Two main ways to connect - Use of mysqli- and PDO (Data Objects)
Making a Connection - Setting with localhost and database name
Reading Data - Use of PDO Data object to fetch the data. Fruit shop application -Demo

Setting Up the Book Club Database

Creating the Book Club Database - Use of UNIQUE Constraint in Book club database for creating database and inserting records

RETRIEVING DATA WITH SELECT

Limit the number of rows returned - Use of select query and its various types
Sorting results - Sort returned rows in any order with orderby query
Summarizing Data - Use of count(), sum(), min(), max(), avg()
Eliminate duplicate results - Use of Distinct keyword in query
Grouping Results - Use of Groupby keyword in query
Pulling Data from Multiple Tables - Use of distinct keyword for combining tables

UNIT V (LECTURE HOURS: 14)

INSERTING , UPDATING & DELETING RECORDS

Inserting a row of data - INSERT query to insert a record
Altering the data within an existing table - UPDATE query to alter a record
Deleting rows of data - DELETE query to delete a record

BUILDING A MEMBER REGISTRATION APPLICATION

Adding more Common code - Defining validateField(), setChecked()
Creating the Registration Script - Creating the register.php file with internal .css and .html file.
Testing the Application - Explanation on register.php script

What Is a Regular Expression?

Syntax for searching for patterns of text within strings - Use delimiters(/),caret (^) character , strstr() function,example

Pattern Matching in PHP

pattern - matching function - preg_match(),function arguments, searching a particular position in the target string

EXPLORING REGULAR EXPRESSION SYNTAX

Matching Literal Characters - Special characters for matching literal characters, escape sequences
Matching Types of Characters using Character Classes - Shorthand character classes,example for character class, Dot to match any character
Matching Multiple Characters - Quantifiers ,example for matching a string of at least one digit
Greedy and Non - Greedy Matching - To match the largest number of characters, to match the smallest number of characters

SEMESTER - V
MAJOR-THEORY
PHP & MY SQL

Using Subpatterns to Group Patterns - preg_match(),$matches with example
Referring to Previous Subpattern Matches - Backreferences, example for referring to previous subpattern matches **Matching Alternative Patterns** -
Explanation on single and double vertical bar operator
Using Anchors to Match at Specified Positions - Assertion, caret (^) symbol, non - word character (!), list of the anchors
Finding Multiple Matches with preg_match_all()
Finding all matches for a regular expression - preg_match_all() function,example for preg_match_all() function
Searching Arrays with preg_grep()
Searching an entire array of strings - Arguments of preg_grep(),example for preg_grep() function
REPLACING TEXT
Replacing Text with preg_replace() - Function str_replace(),Arguments of str_replace()
Replacing Text using a Callback Function - preg_replace_callback(),example to do arithmetic in regular expressions

Text Books :
Beginning PHP 5.3 | Edition:Wiley India Edition | Wrox Publication | Matt Doyle(2012)

Reference Books :
Beginning PHP and MySQL | Edition:Third Edition | From Novice to Professional | W. Jason Gilmore(2008)
PHP and MySQL Web Development | Edition:Fourth Edition | Pearson Publication | Laura Thomson AND Luke Welling(2011) PHP,Apache,MySQL® Web Development | Edition: | Wiley Publishing,Inc | Elizabeth Naramore AND Gary Mailer AND Jason Gerner AND Jeremy Stolz AND Michael Glass AND Yann Le Scouarnec(2004)

SEMESTER - V
ELECTIVE-THEORY
ELECTIVE II - EDC: CYBER SECURITY

OBJECTIVES :

- To understand the threats in cyberspace.
- To know the importance of security in the field of IT .
- To kindle further interest to know more about the chosen field .

HOURS / WEEK - 3		
HOURS / SEMESTER - 45		
CIA	**CO-CURRICULAR ACTIVITIES**	**LECTURE**
6	**3**	**36**

UNIT I (LECTURE HOURS: 7)

Role of Internet& Web Services

Internet - Internet Definition, Benefits of Internet, Limitations

Web Services - Protocols: HTTP , XML , Explanation

Information System Threats - Threat, Vulnerability-Errors & Omissions, Employee Sabotage

Classification of Threats & accessing Damage - Environmental Threat, Denial of Service, Eavesdropping, Malicious Code, Theft & Fraud, Website Intrusion

Components of Threats - Asset, Actor, Motive, Access

UNIT II (LECTURE HOURS: 7)

Security in Mobile And Wireless Computing

Introduction - Introduction , Purpose

Hacking and Cracking - Hackers, Crackers, Explanation

Threats and Attacks Defined - Threats & Attacks Diagrammatic Representation, Explanation.

Accidental Threats - Definition, power failures, loss of communications, water outages and leaks, sewer problems etc **Attacks** - Passive attacks, Active attacks, Covert channels, Dealing with Attacks

UNIT III (LECTURE HOURS: 7)

CYBER CRIMES

Types of Cyber Crimes - Introduction, What is Cyber Crime, Explanation

Cyber Stalking - Definition, Explanation

Cyber Stalker - Definition ,Explanation

How does a Cyber Stalker Operate - Explanation

When does Cyber Stalking happen - Explanation

Denial of Service - Definition, Diagrammatic Representation, Explanation

Purposes of Hacking - Purpose, Explanation

About Hackers, Crackers and Phreaks - Definition, Example with Explanation

Online Fraud - Explanation with Example

Pronography - Explanation with Example

UNIT IV (LECTURE HOURS: 8)

Software Piracy

Examples of Software Piracy - End user copies, Hard disk loading, Counterfeiting, downloads from the Inter net

Spoofing - Definition, Usenet Newsgroup, Explanation

Virus Dissemination - Definition, Explanation

Typical action of a virus - Explanation, Diagrammatic representation of a Virus

World's worst virus attacks - Types Explanation-Love letter, Klez, Melissa, Nimda, Anna Kournikova worm, Asset by Threat, Child Pornography, Cyber Contraband, Cyber Laundering, Cyber Stalking, Cyber terrorism, Cyber theft ,Examples

UNIT V (LECTURE HOURS: 7)

SEMESTER - V
ELECTIVE-THEORY
ELECTIVE II - EDC: CYBER SECURITY

Types of Cyber Crime

Types of Cyber Crime - Spam, Fraud, Obscene or Offensive Content, Harassment, Drug Trafficking, Cyber Terrorism, Cyber Bullying, Piracy

Preventing Cyber Crimes/Safety Measures - General Guidelines on Cyber Safety, Email Safety, Virus Warnings **Protect your Personal Computer** - Important Guidelines, Explanation

Use a Firewall - Software Explanation, Make it a family rule to never give out personal information, Why you should report cyber crime, How to report a Cyber Crime filling a complaint/ Writing an application letter

Text Books :

T1. Information Security And Cyber Laws | Edition:1st Edition | Vayu Education of India | Ankur Shree Aggarwal AND AnuradhaTyagi. Shalu Goel AND Prof.Sanjeev Kumar Sharma( )

Reference Books :

R1. Internet Privacy Kit | Edition: | Que Corporation | Marcus Goncalves(1997)
R2. Internet Security | Edition: | International Thomson Computer Press | OthmarKyas(1997)

SEMESTER - V
SKILL BASED COURSE-THEORY
ADVANCED IP SERVICES

OBJECTIVES :

· Enable the students to learn the advanced principles and techniquesof IP services .
· To understand the concept of advanced IP Configurations , VLAN, STP, OSPF, EIGRP, PPP WAN, ACLs, MISC LAN.
· To understand the concept of networking and able to configure , verify, and troubleshoot complex computer networks.

HOURS / WEEK - 4		
HOURS / SEMESTER - 60		
CIA	**CO-CURRICULAR ACTIVITIES**	**LECTURE**
8	**4**	**48**

UNIT-I (LECTURE HOURS: 10)

Virtual LAN Concepts with Trunks

Virtual LAN Concepts - Definition, Creating two broadcast domains with Switches and VLANs
Creating Multiswitch VLANs using Trunking - VLAN Trunking, Tagging and Identifier **VLAN**
Tagging Concepts - VLAN Trunk, VLAN Trunking between two Switches
The 802.1Q and ISL VLAN Trunking Protocols - 802.1Q Trunk, IEEE 802.1Q and Inter-Switch Link
Forwarding Data Between VLANs - Campus LAN
Routing Packets Between Virtual LAN with a Router - Layer 2 Switch Does Not Route between the VLANs **Routing Packets with a Layer 3 Switch** - Routing between the Two VLANs using Trunk on the Router
Creating VLANs and Assigning Access VLANs to an Interface - How to create VLAN , Give the VLAN a name and Assign interfaces to a VLAN
VLAN Trunking Protocol - Cisco Protocol, VTP Modes
VLAN Trunking Configuration - Type of Trunking, Administrative mode, Switch port mode
Implementing Interfaces Connected to Phones - Data and Voice VLAN Concepts

UNIT-II (LECTURE HOURS: 8)

Spanning Tree Protocol Concepts

Spanning Tree Protocol - Two goals of STP
The Need for Spanning Tree - Prevents three common problems in Ethernet LANs, Broadcast Storm **What IEEE 802.1D Spanning Tree does** - Blocks the port to break the loop
How Spanning Tree Works - Spanning Tree Algorithm, three Criteria
Influencing and Changing the STP Topology - Making configuration changes to influence the STP Topology

OSPF Concepts and Operation

OSPF Overview - Link State protocols builds IP routes
Topology Information and LSAs - LSA and LSDB Relationship
Applying Dijkstra SPF Math to Find the Best Routes - Applying LSA and calculating routes with three phases

Becoming OSPF Neighbors

The Basics of OSPF Neighbors - OSPF Neighbors
Meeting Neighbors and Learning their Router ID - OSPF Hello Process

UNIT-III (LECTURE HOURS: 10)

Understanding EIGRP Concepts

Introduction to EIGRP - Timeline for IG IGPs
Basic Distance Vector Routing - DV or LS routing
Protocol features - Protocols
The concept of a Distance and a Vector - Information learned using DV protocols
Full Update Messages and Split Horizon - Couple of functions to Update Messages and Split Horizon **Route Poisoning** - Route Poisoning

EIGRP Operations

EIGRP Neighbors - Four settings
Exchanging EIGRP Topology Information - EIGRP update Messages

SEMESTER - V
SKILL BASED COURSE-THEORY
ADVANCED IP SERVICES

Calculating the best routes for the Routing Table - Metric calculation
EIGRP Convergence - Feasible distance and Reported distance

UNIT-IV (LECTURE HOURS: 10)

Access Control Lists Basics
ACL Location and Direction - Way to identify the different types of packets
Matching Packets - Location and Direction of ACL
Taking Action When a Match Occurs - Two actions
Types of IP ACLs - ACLs features
Standard Numbered ACLs
List Logic with IP ACLs - List of one or more configuration commands
Matching Logic and Command Syntax - Matching command and Matching Exact IP address, Matching any/all IP Addresses
Advanced Access Control Lists
Extended Numbered IP Access Control Lists - Modifying larger variety of packet header fields
Matching the Protocol, Source IP and Destination IP - IP header, with focus on required fields in extended IP ACLs **Matching TCP and UDP Port Numbers** - IP header followed by a TCP header and Port Number fields **Named IP Access Lists** - Using for filtering Packets
Editing ACLs Using Sequence Numbers - Applying new style for Editing ACLs
Numbered ACL Configuration Vs Named ACL Configuration - Adding and displaying a Numbered ACL Configuration

UNIT-V (LECTURE HOURS: 10)

Miscellaneous LAN
Securing Access with IEEE 802.1x - LAN is built with cable to run each desk
AAA Authentication - How to secure network devices **AAA Login Process** - Install and configure AAA server **TACACA+ and RADIUS Protocols** - Encrypt the passwords **AAA Configuration Examples** - Configure a Router or Switch **DHCP Snooping Basics** - Acts like a firewall or an ACL
An Example DHCP-based Attack - DHCP Attack supplies good IP Address but wrong Default Gateway, DHCP attack leads to Man-in-the-Middle
How DHCP Snooping Works - Summary of Rules for DHCP Snooping Configure one switch rather than an act of multiple switch
Improving Design and Availability with Chassis Aggregation - Key features switch Aggregation

Text Books :
CCNA Routing and Switching ICND2 200-105 Official Cert Guide, CCIE NO.1624 | Edition:3 | Cisco Press | Wendell Odom(2017)

Reference Books :
Computer Networks | Edition:5 | Prentice Hall | Andrew S Tanenbaum( )

SEMESTER - V
MAJOR-THEORY
ANGULARJS

OBJECTIVES :

- Understand the design of single-page applications and how AngularJS facilitates their development .
- Properly separate the model, view and controller layers of application and implement using AngularJS .
- Angular Material is both a UI Component framework and a reference implementation of Google's Material Design Specification.
- It provides a set of reusable, well-tested, and accessible UI components based on Material Design.

HOURS / WEEK - 6		
HOURS / SEMESTER - 90		
CIA	**CO-CURRICULAR ACTIVITIES**	**LECTURE**
12	6	72

UNIT I (LECTURE HOURS: 12)

INTRODUCTION TO ANGULARJS ||ARCHITECTURAL CONCEPTS

Client-side JavaScript framework - Framework, User Interface, Expressive, Reusable,and Maintainable Application Components **Model-View-Controller (MVC)** - MVC,MVW, Language, Platform, and Purpose of the application

What is a directive

Reusable components - An Attribute, Element, Class, and Comment

USING ANGULARJS

BUILT-IN DIRECTIVES

The ngApp directive - The root of an AngularJS application to bootstrap the framework

The ngController directive - The view and controller start to share the same scope and are ready to work together

The ngBind directive - Span element and replaces the content of the element with the results of the provided expression

The ngRepeat directive - Iterate over arrays and objects.The rows of a table, the elements of a list, and the options of select

The ngModel directive - Attaches the element to a property in the scope, binding the view to the model

The ngClick directive and other event directives - Bind any custom behavior to the click event of the element **The ngShow and ngHide directives** - Changes the visibility of an element based on its display property

CREATING OUR OWN DIRECTIVES

Template - The number of times the same snippet of the HTML code repeated over application code **Template Url** - The snippet to an isolated file and bind it using the templateUrl property **Replace** - Discard the original element,replacing it by the directive's template **Restrict** - The directives are restricted to be applied as an attribute to a determined Element

EXPRESSIONS-BASIC USAGE WITH EXPRESSIONS

Date - A date Value comes from the database or any other source in a raw and generic format

Filter - Acting Over an array and applying any filtering criteria

Lowercase - The content of the expression in lowercase

Number - Format a string as a number

Orderby - Order any array based on a predicate expression. String, array, function

Uppercase - The content of the expression in uppercase

UNIT II (LECTURE HOURS: 14)

FORM VALIDATION

Basic validation - The ngRequired directive, to intimate the validation process that the field is actually required

Understanding the $pristine and $dirty properties - Pristine means purity that the field wasn't touched by anyone. After it's been touched for the first time,it becomes dirty

The $error object - Accumulates the detailed list of everything that happens with the form

Dependency injection

The level of Dependency - The object-oriented world, is known as coupling, and indicates the level of Dependency between the components

Creating services

Creating services with the factory - Register the service in the application module that passes two parameters: the name of the service And the factory function

SEMESTER - V
MAJOR-THEORY
ANGULARJS

Creating services with the service - Uses a constructor function, which is equivalent to using the new operator **Creating services with the provider** - The provider relies on the $get function to expose its behavior

Communicating with the backend

HTTP, REST, and JSON - Interact with the backend was Through http with the help of the get and post methods

Creating a single-page application

Installing the module - The $route service by mapping urls against controllers and views, and parameter passing **Configuring the routes** - Controller,templateurl, resolve

Rendering the content of each view - The $route service and is responsible for rendering each template according to the routing mechanism

Passing parameters - Inject the $routeparams service,which will provide us with the parameters passed through the url

THE $ROOTSCOPE OBJECT-SCOPE BROADCASTING

Global Behavior - Injected inside any component such as controllers, directives, filters, and services **$broadcast** - Communicate between components by the means of a scope

UNIT III (LECTURE HOURS: 16)

Introduction

What is angularjs material?& Goals & Principles - UI component framework-Google's material design specification

AngularJS Material Environmental setup

Installing the angularjs material libraries Build - Configuring the library files for dependencies

a material application (blank shell) - Configure the folders and files for application

Introduction Material Design Layouts

Layout and containers - Create modern, responsive layouts on top of CSS3 flexbox

Layout and responsive break points - Associate breakpoints with mediaQuery definitions using breakpoint alias(es):Layout- xs,gt,sm,gt-sm,etc

Layout API and Breakpoint overrides methods - Simple Layout markup convention. The alias is used as suffix extensions to the Layout API keyword

UNIT IV (LECTURE HOURS: 14)

Flex Directive

Responsive flex directives - The flex directive value is restricted to multiples of five,33 or 66

Additional flex values - There are additional flex values provide to improve flexibility and To make the API easier to understand

Ordering HTML elements - Its order position within the layout container. Any integer value from -20 to 20 is accepted

FLEX API and Breakpoint overrides methods - Device width when breakpoint overrides default

Add offsets to the preceding HTML Elements - Flex-offset the margin-left offset is applied

CHILD ALIGNMENT

Set Child alignments within the layout container - The children aligned in the layout's direction and perpendicular to the layout's direction

Layout-Margin - Adds margin around each flex child

Layout-Padding - Adds padding inside each flex child

Layout-wrap - A non-trivial group of flex elements using layout-wrap

Layout-fill - Forces the layout element to fill its parent container

Show & Hide - The show hide APIs to responsively show or hide elements

UNIT V (LECTURE HOURS: 16)

DIRECTIVES

Autocomplete - Search for matches from local or remote data sources

Bottom Sheet - Displayed by click one of the buttons below

Button - Button directive with optional ink ripples

Card - Directive is a container element used within containers

Checkbox - Directive is used like the normal angular checkbox

Chips - Component used within and is responsible for rendering individual chips

Content - Directive is a container element useful for scrollable content

Dialog - The dialog's template must be inside this element

Datepicker - The dialog's template must be inside this element

Divider - Dividers group and separate content within lists and page layouts using strong visual and spatial distinctions

SEMESTER - V
MAJOR-THEORY
ANGULARJS

FAB Toolbar - Directive is used to present a toolbar of elements for quick access to common actions when a floating action button is activated

Input - Build complex forms for data entry

Icon - Directive makes it easier to use vector-based icons in app

Input-Container - The parent of any input or textarea element

List - Directive is a list container for 1..n tags

Nav Bar - Directive renders a list of material tabs that can be used for top-level page navigation **Panel** - Create dialogs, menus, and other overlays

Radio Button - Directive is the child directive required to be used within elements

Sidenav - Component that can be opened and closed programatically

Slider - Components allows the user to choose from a range of values

Select - Component can be used within a or as a stand alone component by using the md-no-underline class

Switch - Enable or disable based on the expressions

Subheader - Directive creates a sticky subheader for a section

Toolbar - Place a toolbar in your app

Toast - Toast can be dismissed with a swipe, a timer, or a button

Tabs - Specify a tab with a label and optional view content

Whiteframe - Apply an elevation shadow to an element

User-card - Cards avatar,Class for user image

Text Books :

Angular JS Essential | Edition: | Packt Publication. | Rodrigo Branas( )

SEMESTER-V
MAJOR-X
RELATIONAL DATABASE MANAGEMENT SYSTEM

OBJECTIVES:

1. To enable the students to understand the concepts of Database Management System
2. To understand the Normalization process and ER modeling
3. To create various queries to solve complex scenarios

PEDAGOGY:

- Lecture, Group Discussion, LCD, OHP, Seminar & Case Study.

TOTAL HOURS/WEEK – 5				
TOTAL HOURS/SEMESTER - 75				
CIA	**CO-CURRICULAR ACTIVITIES**	**LECTURE**	**TUTORIAL**	**TOTAL**
10	**5**	**48**	**12**	**75**

Unit I- Lecture Hours : 8 Hrs Tutorial Hours: 2 Hrs				
TOPIC(S)	**SUB TOPIC(S)**	**MINUTES/SUB TOPIC**	**KEY POINT(S)**	**HOURS /TOPIC**
Introduction to database concepts	Introduction	60	Basic building block- collection of facts, Qualitative Vs Quantitative, Example: What do we know about the Dog?	4
	Database		Definition	
	Database Management System		Definition and purpose	

	Database Applications	60	Enterprise , banking and Finance, characteristics of a good DB	
	File processing systems	60	Characteristics of a good database, Data Storage relationship, Isolated, Example	
	Advantages over file systems		• Reduces Data redundancy and data inconsistency • Data isolation • Data integrity • Atomicity of operations • Concurrency • Security	
	Data Abstraction	60	View of Data, University Database Example	
Relational Data Model	Introduction	60	Attributes, Domain, Example	1
	Database Schema and Instance		Definition Relational Schema, Relational instance Example	
	Physical Data Independence		Definition, Example	
	Alternative Terminology			
Keys	Introduction	60	Purpose of the keys	3
	Super Key		Definition, Example	
	Candidate key			

	Primary key	60	Points to remember for a Primary key	
	Secondary Key		Definition, Example	
	Foreign key			
	Query Languages	60	Procedural language Nonprocedural language Definition, Example	
Tutorial Hour	Keys	120	Defining Various key from a database.	2
Unit II- Lecture Hours : 9 Hrs Tutorial Hours: 2 Hrs				
Basic Structure SQL	Introduction	60	Declarative language Example	4
	Several types of SQL		DDL, DML, Integrity, View, Transaction, Embedded SQL, Authorization	
	Select clause	60	Syntax and Examples	
	From clause	60		
	Where clause	60		
Creating and	Create table	60	Syntax and example	2

Renaming a relation	The Rename Operation	60	Syntax and example	
	Tuple Variables		Syntax and example	
String Operations	Like	60	Percent (%) Underscore (_)	1
Display of Tuples	order by clause	60	Ascending and descending order	2
	Set operations	60	Union Operation ,Union all, Intersect Operation, Except Operation	
Tutorial Hour	Table Definition	120	Basic of table creation and clauses	2
Unit III- Lecture Hours : 12 Hrs Tutorial Hours: 3 Hrs				
Aggregate and Grouping Functions	Basic Aggregation	60	Average, Sum ,Count, Maximum, Minimum	3
	group by	60	Syntax and example	
	Having	60	Syntax and example	
	Insertion	60	Syntax and example	4
	Deletion	60		
	Updates	60		

Modification of the Database				
	Data types	60	Standard and Other	
Join Expressions	Join types and Join conditions	60	Syntax and example	3
	natural join operation		Syntax and example	
	Outer join	60	Left right, full outer join	
	View Definition	60	Syntax and example	
Constraints	Not null constraint	60	Syntax and Example	2
	Unique Constraint		Syntax and Example	
	The check Clause		Syntax and Example	
	Default Values			
	Primary key constraint	60	Syntax and Example	
	Referential Integrity		Syntax and Example	
	alter table		Add, Modify, drop table	
Tutorial Hour	Sql operations	180	Querying	3
Unit IV- Lecture Hours : 11 Hrs Tutorial Hours: 3 Hrs				
Relational Database Design	Features of Good Relational Designs	60	Scenarios and Examples	1
	Introduction		FDs are domain knowledge - DB engine will not help, no	

Functional Dependency (FD)		60	optimization, Relation satisfying a dependency Vs Dependency holding on a schema	4
	Types of FD		Trivial, Non trivial, Completely non-trivial	
	Minimal FDs	60	Definition, Example	
	Dependency Preservation	60	Definition, Example	
	Closure and covers of set of functional dependencies		Definition, Example	
	Armstrong's axioms	60	Reflexive, Augmentation Transitive Decomposition Union Pseudotransitivity	
Basic Normal forms	First Normal Form (1NF)	60	1NF: based on attributes only	3
	Second Normal Form (2NF)	60	Prime attribute, full functional dependency and partial dependency	
	Third Normal Form (3FN)	60	Transitive Dependency	
	Boyce-Codd normal (BCNF)	60	Normal forms Lossless decomposition	3

Higher Normal Forms			Anomalies with BCNF	
	Fourth normal form (4NF)	60	Multi-valued dependency (MVD) MVD and lossless join	
	Fifth normal form (5NF) or Project-Join normal form (PJNF)	60	Join dependency (JD)	
	Domain-Key normal form (DKNF)		Syntax and Example	
Tutorial Hour	Normalization	180	Complete Example	3
Unit V- Lecture Hours : 8 Hrs Tutorial Hours: 2 Hrs				
The ER Model constraints	Entity Sets	60	Definition and example	2
	Relationship Sets			
	Attributes			
	Mapping Cardinalities	60	Definition and example	
	Participation Constraints			
	Keys			
	What to remove			
Entity-Relationship Diagrams	Basic Structure	60	Example	2
	Mapping Cardinality			

	Complex Attributes			
	Roles	60	Example	
	Non binary Relationship Sets			
	Weak Entity Sets			
	E-R diagram for the University Enterprise		Example – diagram	
Reduction to Relational Schemas	Representation of Strong Entity Sets with Simple Attributes	60	Example	2
	Representation of Strong Entity Sets with Complex Attributes		Example	
	Representation of Weak Entity Sets	60	Example	
	Representation of Relationship Sets		Redundancy of Schemas	
			Combination of Schemas	
Extended E-R Features:	Specialization	120	Example	2
	Generalization		Example	
	Inheritance		Example	
Tutorial Hour	ER Diagram Creation	120	Example	2

Text Book:

1. Database System Concepts, Silberschatz-Korth-Sudarshan, 6th Edition, 2011, TMH Publications

Reference Books:

1. Database System Concepts, Silberschatz-Korth-Sudarshan, 4th Edition, 2001, TMH Publications.

2. Fundamentals of Database Systems, Ramez Elmasri, Shamkant B. Navathe, 6th Edition , 2011, Pearson Education publication.

3. Beginning Database Solutions, Rod Stephens, 2009, Wrox Publication

SEMESTER – V

MAJOR PAPER - XI

PHP & My SQL

OBJECTIVES:

- Understand and demonstrate analytical powers of thought through critical analysis, evaluation and synthesis of fundamentals of PHP language and its associated Technologies.
- Demonstrate awareness on using PHP to create dynamic interactive Web forms.
- To create flexibility in designing, planning and building real-world projects.
- Develop specific focus and depth of critical understanding and interpretation in specific areas in the Web Development

PEDAGOGY:

- Teaching aids used are Black board, OHP, Projector, Demonstration and Group Discussion.

TOTAL HOURS/WEEK – 5			
TOTAL HOURS/SEMESTER – 75			
CIA	**CO-CURRICULAR ACTIVITIES**	**LECTURE**	**TUTORIAL**
10	**5**	**60**	**-**

UNIT : I (LECTURE HOURS : 12				
TOPIC(S)	**SUB TOPIC(S)**	**MINUTES**	**KEY POINT(S)**	**HOURS**
What is PHP? & Why Use PHP ?	A dynamic web page	60	Interactive web site	2
	Process of running PHP script		Clicking a link, requested URL, Script runs	
	Common examples of PHP scripts		Web forum, Online stores, Search enginees,Blogs,Web mail applications	
	Feature of PHP	60	Cross Platform, Integrate with all web servers	
	How does PHP compare with other common Web programming technologies?		Asp,Asp.Net,Perl,Java,Python,Ruby, ColdFusion	
The Evolution of PHP & What's New in PHP 5.3	Early history of PHP	60	PHP version 2,PHP version 3.3,PHP 4,PHP 5	1
	Namespaces		Avoid naming clashes	
	The goto Operator		Example	
	Nowdoc Syntax		Embed a block of PHP code	
	Shorthand Form of the Ternary Operator		Syntax	

	Advanced Changes		Late static bindings, Optional garbage collector	
Installing PHP	Installing on Ubuntu Linux	60	Steps to install on Ubuntu Linux	2
	Installing on Windows		Steps to install WampServer	
	Installing on Mac OSX		Steps to install on Mac OS	
	Testing Your Installation	60	Testing the Web Server	
	Testing PHP		Open text editor,type port number,phpinfo()	
	Setting Your Time Zone		Steps to set your time zone	
Other Ways to Run PHP	Running PHP with other Web Servers	60	Internet Information Server, ISAPI module	1
	Compiling PHP Yourself		Basic steps to compile PHP	
	Running PHPRemotely		Ssh package	
Creating Your First Script	Embedding PHP within HTML	60	Code to create a Stylish Page, embed PHP within an HTML page	1
	Enhancing the Script Further		Example to enhance the script to display the current time	
	Using Comments to Make Code More Readable		Single line comments and multi - line comments	
Using Variables in PHP &	Naming Variables	60	Rules for naming variables, example	1

Understanding Data Types	Creating Variables		Initializing a variable, example of declaring and initializing a variable	
	About Loose Typing		Loosely - typed language,example	
	Testing the Type of a Variable		gettype() function, PHP ' s type testing functions	
	Changing a Variable's Data Type		settype() function,example	
	Changing Type by Casting		List of casting functions that used in PHP	
Operators and Expressions	Operator Types	60	Arithmetic, assignment, logical, comparison, error control, incrementing-decrementing, Array, string	1
	Understanding Operator Precedence.		Precedence of Some PHP Operators (Highest First)	
	Constant values		define() function	
Making Decisions	Simple Decisions with the if statement	60	if syntax and its example	1
	Providing an Alternative Choice with the else statement		if…else, else…if syntax and its example	
	Testing one Expression Many Times with the switch statement		switch statement- syntax and its example	

	Compact coding with the Ternary Operator		ternary operator syntax and its example	
Doing Repetitive Tasks with Looping	Simple Looping with the while statement	60	while syntax with example for loops general syntax and its example	2
	Testing at the End: The do…while Loop		do…while syntax with example	
	Neater Looping with the for statement		The general syntax of a for loop	
	Escaping from Loops with the break statement		Explanation on break statement with an example	
	Skipping Loop iterations with the continue statement		Explanation on continue statement with an example	
	Creating Nested Loops		A Homing Pigeon Simulator :Application	
	Mixing Decisions and Looping with HTML	60	Fibonacci sequence example	
UNIT : II (LECTURE HOURS: 12)				
Creating and Accessing Strings	Including More Complex Expressions within Strings	60	Common escape sequences	1

	Using Your Own Delimiters		heredoc syntax and nowdoc syntax	
	Other ways to create Strings		PHP function return string values	
	Finding the Length of a String		strlen() , str_word_count() functions	
	Accessing Characters within a String		About substr() function	
Searching and replacing text within Strings	Searching Strings with strstr()	60	Syntax and usage of various string search function with example	2
	Locating Text with strops() and strrpos()			
	Finding the Number of Occurrences with substr_count()			
	Searching for a Set of characters with strpbrk()			
	Replacing All Occurrences using str_replace()	60	Syntax for str_replace(),substr_replace() and strstr() syntax with example	
	Replacing a Portion of a String with substr_replace()			

	Translating Characters with strstr()			
Dealing with Upper- and Lowercase: **Formatting Strings**	String functions for converting upper and lower cases		Syntax for strtolower(), strtoupper(), ucfirst(), lcfirst() functions	2
	General- Purpose Formatting with printf() and sprint()	60	printf() versus sprintf(),Using type specifiers, Specifying Signs, Padding the Output, Specifying Number Precisions, Swapping arguments, Storing the Result Instead of Printing it	
	Trimming Strings with trim(),ltrim() and rtrim()	60	Purpose of trim functions and its syntax	
	Padding Strings with str_pad()		Adding space to the left and right of the string	
	Wrapping Lines of Text with wordwrap()		Splitting of lines to several lines using wordwrap()	
	Formatting Numbers with number_format()		Purpose of number_format function and its syntax	
Creating & **Accessing Array Elements**	Two types of array	60	Indexed arrays , Associative arrays,how does PHP arrays work	1
	Creating a new array variable		built-in array() construct	
	Changing Elements		Manipulating pointer using count() print_r(), array_slice(), each(),	

	Outputting an Entire Array with print_r() Extracting a Range of Elements with array_slice() Counting Elements in an Array Stepping Through an Array		current(), key(), next(), prev(), end() and reset() function and its descriptions.	
Looping Through Arrays with foreach	Using foreach to Loop Through values Using foreach to Loop Through Keys and Values Altering Array Values with foreach	60	Syntax for - foreach… loop and a demo program to retrieve key and value pairs.	1
Working with Multidimensional Arrays	Creating a Multi-dimenstional Array Accessing Elements of Multidimensional Arrays Looping through Multidimensional Arrays	60	Demonstration of displaying an array of Books application	1
	Definition	60	Subroutine,calling code	

What Is a Function? & Why Functions are Useful	Reasons to use functions		Avoid duplicating code, easy to eliminate errors,reuseablitity	1
Calling & Working with Variable Functions	A function call	60	Syntax for calling function and example	1
Writing Your Own Functions	Defining Parameters	60	Use of function with defined parameters and default values	2
	Optional Parameters and Default values			
	Returning Values from your Functions		Understanding local and global variable.	
	Understanding Variable Scope			
	Writing Recursive Functions	60	Definition,base case,recursive case,overview of recursive function, Creating the Fibonacci Sequence with Recursion	
UNIT : III (LECTURE HOURS: 12)				
How HTMLForms Work	Form tag with its attributes	60	Action,method,example, How it work form fields, form elements with get and post method	1

Capturing Form Data with PHP	Superglobal arrays	60	$_GET, $_POST, $_REQUEST	1
Dealing with Multi-Value Fields & Generating Web Forms with PHP	Adding multi-value fields in php script	60	adding square brackets ([]),example two common approaches to generate form within PHP	1
Saving State with Query Strings	Building & Accessing Data Query Strings	60	Query string,query string characters t used within field names and values, http_build_ query() function, $_GET superglobal array	1
Working with Cookies	Cookie Components	60	What is cookies,example to create a cookie,fields in cookie	1
	Setting a Cookie in PHP		setcookie()function,example to use setcookie()	
	Accessing & Removing Cookies in Your Scripts		$_COOKIE superglobal array,example	
Using PHP Sessions to Store Data	Creating a Session	60	session_ start() function,	1
	Reading and Writing Session Data		$_SESSION[]	
	Destroying a Session		session_destroy() function, $_SESSION array, session_name()	
	Passing Session I in Query Strings		PHPSESSID field, session_id() function	

	Changing Session Behavior		Directives in php.ini file	
Understanding Files and Directories	File & Directories	60	Importance of files and directories	1
Opening and Closing Files	Opening a File with fopen()	60	fopen() function usage and explanation	1
	Closing a File with fclose()		fclose() function usage and explanation	
Reading and Writing to Files	Functions to read and write the files	60	About fread(),fwrite(),fseek(),feof(),fgets(), fgetc()	1
Working with Dates and Times	Getting the Current date and time	60	date(), getdate(), setdate(), checkdate(), time(), mktime()	1
Creating and Manipulating Images	Creating a New image	60	About imagecreate()	2
	Allocating colors		imagecolorallocate(), imagecolorresolve(),imagejpeg(), imagegif(),imagepng(),imagesetpixel(),imageline(),imageline(), imageellipse(),imagerectangle(), imagepolygon(), imagearc()	
	Outputting Images	60	imagecreatefromjpeg(),imagecreatefromgif(),imagecreatefrompng()	
	Drawing in an image			

	Opening an Existing Image			
UNIT : IV (LECTURE HOURS: 12				
How to Handle Exceptions	How to create and throw exceptions	60	The syntax for a try-catch statement, A try-catch statement that catches an exception object, A try-catch statement that throws an exception object, A try-catch statement that catches two types of exception ,description	1
	How to use the try-catch statement		The syntax for creating a new exceptions, The syntax for the throw an exception, A statement that causes an exception to be thrown, methods of exception objects	
Setting Up MySQL	Starting the MySQL Server	60	The MySQL server and its command-line tool, Ubuntu-WAMP Server on Windows, MAMP on Mac OS X	1
	Setting Up the MySQL root Password		About root directory, root password, setting privileges	
A Quick Play with MySQL	Creating a New Database	60	Explaination on CREATE DATABASE command,use command,create table,show tables, Insert,Update,Delete,Drop Commands and its examples	5
	Creating a Table			
	Adding Data to a Table	60		
	Reading Data from a Table	60		
	Updating Data in a Table	60		

	Deleting Data from a Table	60		
	Deleting Tables and Databases			
Connecting to MySQL from PHP	Two main ways to connect	60	Use of mysqli- and PDO (Data Objects)	1
	Making a Connection		Setting with localhost and database name	
	Reading Data		Use of PDO Data object to fetch the data. Fruit shop application –Demo	
Setting Up the Book Club Database	Creating the Book Club Database	60	Use of UNIQUE Constraint in Book club database for creating database and inserting records	1
Retrieving Data with SELECT	Limit the number of rows returned	60	Use of select query and its various types	3
	Sorting results		Sort returned rows in any order with orderby query	
	Summarizing Data	60	Use of count(), sum(), min(), max(), avg()	
	Eliminate duplicate results		Use of Distinct keyword in query	
	Grouping Results	60	Use of Groupby keyword in query	
	Pulling Data from Multiple Tables		Use of distinct keyword for combining tables	
UNIT : V (LECTURE HOURS: 12)				
	Inserting a row of data	60	INSERT query to insert a record	1

Inserting , Updating & Deleting records	Altering the data within an existing table		UPDATE query to alter a record	
	Deleting rows of data		DELETE query to delete a record	
Building a Member Registration Application	Adding more Common code	180	Defining validateField(), setChecked()	3
	Creating the Registration Script		Creating the register.php file with internal .css and .html file.	
	Testing the Application		Explanation on register.php script	
What Is a Regular Expression?	Syntax for searching for patterns of text within strings	60	Use delimiters(/),caret (^) character , strstr() function,example	1
Pattern Matching in PHP	pattern - matching function	60	preg_match(),function arguments, searching a particular position in the target string	1
Exploring Regular Expression Syntax	Matching Literal Characters	180	Special characters for matching literal characters, escape sequences	3
	Matching Types of Characters using Character Classes		Shorthand character classes,example for character class, Dot to match any character	
	Matching Multiple Characters		Quantifiers ,example for matching a string of at least one digit	
	Greedy and Non - Greedy Matching		To match the largest number of characters, to match the smallest number of characters	

	Using Subpatterns to Group Patterns		preg_match(),$matches with example	
	Referring to Previous Subpattern Matches		Backreferences, example for referring to previous subpattern matches	
	Matching Alternative Patterns		Explanation on single and double vertical bar operator	
	Using Anchors to Match at Specified Positions		Assertion, caret (^) symbol, non - word character (!), list of the anchors	
Finding Multiple Matches with preg_match_all()	Finding all matches for a regular expression	60	preg_match_all() function,example for preg_match_all() function	1
Searching Arrays with preg_grep()	Searching an entire array of strings	60	Arguments of preg_grep(),example for preg_grep() function	1
Replacing Text	Replacing Text with preg_replace()	60	Function str_replace(),Arguments of str_replace()	1
	Replacing Text using a Callback Function		preg_replace_callback(),example to do arithmetic in regular expressions	

TEXT BOOKS:

T1. Matt Doyle, Beginning PHP 5.3, Wrox Publication,Wiley India Edition, 2012.

Chapters 1,2,3,4,5,6,7,9,10,11,13,14,18

REFERENCE BOOKS:

R1. W. Jason Gilmore, Beginning PHP and MySQL: From Novice to Professional, Third Edition, 2008.

R2. Michael Glass,Yann Le Scouarnec,Elizabeth Naramore,Gary Mailer,Jeremy Stolz,Jason Gerner,

PHP, Apache,MySQL® Web Development,Wiley Publishing,Inc,2004.

R3. Luke Welling, Laura Thomson, PHP and MySQL Web Development, Pearson Publication, Fourth Edition,2011.

R4. Joel Murach and Ray Harris, Murach's PHP & MySQL, Murach publication,3rd edition. **Chapter:** 15 (Exception Handling) Page 482-486.

SEMESTER - V
MAJOR PAPER-XII
AngularJS

OBJECTIVES:

- & Understand the design of single-page applications and how AngularJS facilitates their development.
- & Properly separate the model, view and controller layers of application and implement using AngularJS.
- & Angular Material is both a UI Component framework and a reference implementation of Google's Material Design Specification.
- & It provides a set of reusable, well-tested, and accessible UI components based on Material Design.

PEDAGOGY:

2. **Teaching aids used are Black board, OHP, Projector, Demonstration and Group Discussion.**

TOTAL HOURS/SEMESTER - 90		
CIA	**CO-CURRICULAR ACTIVITIES**	**LECTURE**
12	**6**	**72**

UNIT : I (LECTURE HOURS:12)				
TOPIC(S)	**SUB TOPIC(S)**	**MINUTES**	**KEY POINT(S)**	**HOURS**
Introduction to AngularJS	Client-side JavaScript framework	60	Framework, User Interface, Expressive, Reusable,and Maintainable Application Components	1
Architectural concepts	Model-View-Controller (MVC)		MVC,MVW, Language, Platform, and Purpose of the application	
What is a directive	Reusable components	60	An Attribute, Element, Class, and Comment	1

Using AngularJS built-in directives	The ngApp directive	120	The root of an AngularJS application to bootstrap the framework	2
	The ngController directive		The view and controller start to share the same scope and are ready to work together	
	The ngBind directive		Span element and replaces the content of the element with the results of the provided expression	
	The ngRepeat directive	60	Iterate over arrays and objects.The rows of a table, the elements of a list, and the options of select	1
	The ngModel directive		Attaches the element to a property in the scope, binding the view to the model	
	The ngClick directive and other event directives		Bind any custom behavior to the click event of the element	
	The ngShow and ngHide directives		Changes the visibility of an element based on its display property	
Creating our own directives	Template	120	The number of times the same snippet of the HTML code repeated over application code	2
	Template Url		The snippet to an isolated file and bind it using the templateUrl property	
	Replace	60	Discard the original element,replacing it by the directive's template	1
	Restrict		The directives are restricted to be applied as an attribute to a determined Element	
Expressions- **Basic usage with expressions**	Date	120	A date Value comes from the database or any other source in a raw and generic format	2
	Filter		Acting Over an array and applying any filtering criteria	
	Lowercase	120	The content of the expression in lowercase	2
	Number		Format a string as a number	

	Orderby		Order any array based on a predicate expression. String, array, function	
	Uppercase		The content of the expression in uppercase	
UNIT : II(LECTURE HOURS:14)				
TOPIC(S)	**SUB TOPIC(S)**	**MINUTES**	**KEY POINT(S)**	**HOURS**
Form validation	Basic validation	120	The ngRequired directive, to intimate the validation process that the field is actually required	2
	Understanding the $pristine and $dirty properties		Pristine means purity that the field wasn't touched by anyone. After it's been touched for the first time,it becomes dirty	
	The $error object		Accumulates the detailed list of everything that happens with the form	
Dependency injection	The level of Dependency	120	The object-oriented world, is known as coupling, and indicates the level of Dependency between the components	2
Creating services	Creating services with the factory	60	Register the service in the application module that passes two parameters: the name of the service And the factory function	1
	Creating services with the service	120	Uses a constructor function, which is equivalent to using the new operator	2
	Creating services with the provider		The provider relies on the $get function to expose its behavior	
Communicating with the backend	HTTP, REST, and JSON	60	Interact with the backend was Through http with the help of the get and post methods	1
Creating a single-page application	Installing the module	120	The $route service by mapping urls against controllers and views, and parameter passing	2

	Configuring the routes	60	Controller,templateurl, resolve	1
	Rendering the content of each view	120	The $route service and is responsible for rendering each template according to the routing mechanism	2
	Passing parameters		Inject the $routeparams service, which will provide us with the parameters passed through the url	
The $rootScope object-Scope Broadcasting	Global Behavior	60	Injected inside any component such as controllers, directives, filters, and services	1
	$broadcast		Communicate between components by the means of a scope	

UNIT : III(LECTURE HOURS:16)

TOPIC(S)	SUB TOPIC(S)	MINUTES	KEY POINT(S)	HOURS
Introduction	What is angularjs material?& Goals & Principles	120	UI component framework-Google's material design specification	2
AngularJS Material Environmental setup	Installing the angularjs material libraries	180	Configuring the library files for dependencies	3
	Build a material application (blank shell)	180	Configure the folders and files for application	3
Introduction Material Design Layouts	Layout and containers	180	Create modern, responsive layouts on top of CSS3 flexbox	3
	Layout and responsive break points		Associate breakpoints with mediaQuery definitions using breakpoint	

		120	alias(es):Layout-xs,gt,sm,gt-sm,etc	2
	Layout API and Breakpoint overrides methods	180	Simple Layout markup convention. The alias is used as suffix extensions to the Layout API keyword	3

UNIT : IV(LECTURE HOURS:14)

TOPIC(S)	SUB TOPIC(S)	MINUTES	KEY POINT(S)	HOURS
Flex Directive	Responsive flex directives	120	The flex directive value is restricted to multiples of five,33 or 66	2
	Additional flex values	120	There are additional flex values provide to improve flexibility and To make the API easier to understand	2
	Ordering HTML elements	60	Its order position within the layout container. Any integer value from -20 to 20 is accepted	1
	FLEX API and Breakpoint overrides methods	120	Device width when breakpoint overrides default	2
	Add offsets to the preceding HTML Elements	60	Flex-offset the margin-left offset is applied	1
Child Alignment	Set Child alignments within the layout container	180	The children aligned in the layout's direction and perpendicular to the layout's direction	3
	Layout-Margin		Adds margin around each flex child	
	Layout-Padding		Adds padding inside each flex child	

	Layout-wrap		A non-trivial group of flex elements using layout-wrap	
	Layout-fill		Forces the layout element to fill its parent container	
	Show & Hide	180	The show hide APIs to responsively show or hide elements	3

UNIT : V(LECTURE HOURS:16)

TOPIC(S)	SUB TOPIC(S)	MINUTES	KEY POINT(S)	HOURS
Directives	Autocomplete		Search for matches from local or remote data sources	
	Bottom Sheet	60	Displayed by click one of the buttons below	1
	Button		Button directive with optional ink ripples	
	Card	120	Directive is a container element used within <md-content> containers	2
Directives	Checkbox		Directive is used like the normal angular checkbox	
	Chips		Component used within <md-chips> and is responsible for rendering individual chips	
	Content	120	Directive is a container element useful for scrollable content	2
	Dialog	120	The dialog's template must be inside this element	2
	Datepicker		The dialog's template must be inside this element	

	Divider		Dividers group and separate content within lists and page layouts using strong visual and spatial distinctions	
	FAB Toolbar	60	Directive is used to present a toolbar of elements for quick access to common actions when a floating action button is activated	1
	Input		Build complex forms for data entry	
	Icon	120	Directive makes it easier to use vector-based icons in app	2
	Input-Container		The parent of any input or textarea element	
Directives	List		Directive is a list container for 1..n <md-list-item> tags	
	Nav Bar		Directive renders a list of material tabs that can be used for top-level page navigation	
	Panel	120	Create dialogs, menus, and other overlays	2
	Radio Button		Directive is the child directive required to be used within <md-radio-group> elements	
	Sidenav		Component that can be opened and closed programatically	
	Slider	60	Components allows the user to choose from a range of values	1
	Select		Component can be used within a <md-input-container> or as a stand alone component by using the md-no-	

Directives			underline class	
	Switch		Enable or disable based on the expressions	
	Subheader	60	Directive creates a sticky subheader for a section	1
	Toolbar		Place a toolbar in your app	
	Toast		Toast can be dismissed with a swipe, a timer, or a button	
		60		1
	Tabs		Specify a tab with a label and optional view content	
	Whiteframe		Apply an elevation shadow to an element	
	User-card	60	Cards avatar,Class for user image	1

TEXT BOOKS:

Angular JS Essential by Rodrigo Branas, Packt Publication.

Reference Website:
https://material.angularjs.org
https://www.codeschool.com

SEMESTER V

MAJOR PRACTICAL (MP-V)

PROGRAMMING LAB-V RELATIONAL DATABASE MANAGEMENT SYSTEM & PHP & MYSQL

COMPUTER SCIENC/INFORMATION TECHNOLOGY

PHP & MYSQL

UNIT-I

1. Display greeting message based on IST timezone

Design a PHP program to greet the users, who visit the homepage of a particular website according to current timein PHP. The date function with parameter 'H' is used to find Current Time. This program works according to ISTtime zone. (a)If the user enters the time less than 12, it displays a greeting message as "Good Morning" followed byrelevant quotes and greeting images. (b)If the user enters the time between 12 to 17, it displays a greeting messageas "Good Afternoon" followed by relevant quotes and greeting images. (c)If the user enters the time greater than 18,it displays a greeting message as "Good Night", followed by relevant quotes and greeting images. HINT: Use date/timefunction to fetch the instant time from the server where your PHP script runs.

2. Product Ranking Syste

Build a web page that performs a product rating, of latest mobile phones available in the market. The product ratingis carried out using the grade scale like good, very good, excellent in a web form. The user must select any one ofthe grading scale according to the various features available in each product like sound, battery, video, display,cost, etc. The user can also provide their own suggestions to improve branding.HINT: $_Post is a variable used to fetchthe form elements values.

UNIT-II

3. Multi-dimensional Array

Rewrite the large cities array into a multi-dimensional array called $multiCity. The first sub-array will be completelynew and include the labels, City, Country, Continent. Each of the succeeding sub-arrays should include those threeitems, one for each of the cities, for a total of 6 sub-arrays.Here's the

content for your array: City, Country,Continent; Tokyo, Japan, Asia; New York City, USA, North America; Mumbai, India, Asia; Shanghai, China, Asia;London, UK, Europe.In the HTML, use the array in a table. The first row should be a header row and contain theentries in the first sub-array. Call these items without using a loop. For the succeeding rows, use a for loop with aforeach loop nested inside to populate the table with the remaining contents of the array. Use the count() functionso that your for loop will function properly even if you increment or decrement the array. HINT: Add a simple style sheet inthe head section of your HTML.

4.Pragathi store database manipulations

Create a database_table named pragathi_store and make various entries like item _number, item_name, quantity,unit_cost, total _purchase_cost. Use PhpMyAdmin to insert more than 5 records.Performupdation and deletion ofrecords based on relevant constraints. HINT: 'Edit item_name based on item_number for a particular record' is one of the example constraints that can be setforth in query.

UNIT-III

5.Display Customer Preferences from MySQL in Table Format

Code a PHP program that fetch the customer information like Cust_no, Cust_name, Item_purchased and Mob_nofrom the MySQL database table. Use mysql_connect() to connect

from PHP to MySQL and mysql_query() to fetch allthe records from a database table named "customer". Before starting with PHP code create a table with all therelevant fields in PHPMyAdmin.HINT: Display all the information about a customer in table format on the web page.

6.Feedback system

Create a simple webpage about cosmetic products by inserting images, video files and a feedback form, fetchvaluable suggestions from the customer end. Create a table named cosmetics with various fields like name, age,gender, phone no and feedback before running the source code of PHP. HINT:Use .css to create a stylesheet and storethe feedbacks into database table using MySQL.

RDBMS

Unit IV

Problem Domain

• The bank is organized into branches. Each branch is located in a particular city and is identified by a unique name. The bank monitors the assets of each branch.

• Bank customers are identified by their *customer-id* values. The bank stores each customer's name, and the street and city where the customer lives. Customers may have accounts and can take out loans. A customer may be associated with a particular banker, who may act as a loan officer or personal banker for that customer.

• Bank employees are identified by their *employee-id* values. The bank administration stores the name and telephone number of each employee, the names of the employee's dependents, and the *employee-id* number of the employee's manager. The bank also keeps track of the employee's start date and, thus, length of employment.

• The bank offers two types of accounts—savings and checking accounts. Accounts can be held by more than one customer, and a customer can have more than one account. Each account is assigned a unique account number. The bank maintains a record of each account's balance, and the most recent date on which the account was accessed by each customer holding the account. In

addition, each savings account has an interest rate, and overdrafts are recorded for each checking account.

• A loan originates at a particular branch and can be held by one or more customers. A loan is identified by a unique loan number. For each loan, the bank keeps track of the loan amount and the loan payments. Although a loan payment number does not uniquely identify a particular payment among those for all the bank's loans, a payment number does identify a particular payment for a specific loan. The date and amount are recorded for each payment.

1. Creating and renaming a relation
2. Constraints
3. Modification of the Database
4. String Operations
5. Display of Tuples
6. Aggregate and Grouping Functions
7. Join Expressions
8. Sub

SEMESTER –V

ELECTIVE II (EL-II)

EDC-CYBER SECURITY

OBJECTIVES:

- To understand the threats in cyberspace.
- To know the importance of security in the field of IT.
- To kindle further interest to know more about the chosen field.

PEDAGOGY:

- Teaching aids used are Black board, OHP, Projector, Demonstration and Group Discussion.

TOTAL HOURS/WEEK – 3			
TOTAL HOURS/SEMESTER – 45			
CIA	**CO-CURRICULAR ACTIVITIES**	**LECTURE**	**TUTORIAL**
6	**3**	**36**	**-**

UNIT : I (LECTURE HOURS: 7)

TOPIC(S)	SUB TOPIC(S)	Minutes	KEY POINT(S)	Hours
Role of Internet& Web Services	Internet	60	Internet Definition, Benefits of Internet, Limitations	1
	Web Services	120	Protocols: HTTP , XML , Explanation	2
	Information System Threats	60	Threat, Vulnerability-Errors & Omissions, Employee Sabotage	1
	Classification of Threats & accessing Damage	120	Environmental Threat, Denial of Service, Eavesdropping, Malicious Code, Theft & Fraud, Website Intrusion	2
	Components of Threats	60	Asset, Actor, Motive, Access	1

UNIT : II (LECTURE HOURS: 7)				
Security in Mobile And Wireless Computing	Introduction	60	Introduction , Purpose	1
	Hacking and Cracking	120	Hackers, Crackers, Explanation	2
	Threats and Attacks Defined		Threats & Attacks Diagrammatic Representation, Explanation.	
	Accidental Threats	120	Definition, power failures, loss of communications, water outages and leaks, sewer problems etc	2
	Attacks	120	Passive attacks, Active attacks, Covert channels, Dealing with Attacks	2
UNIT : III (LECTURE HOURS: 7)				
Cyber Crimes	Types of Cyber Crimes	120	Introduction, What is Cyber Crime, Explanation	2
	Cyber Stalking		Definition, Explanation	

	Cyber Stalker	120	Definition ,Explanation	2
	How does a Cyber Stalker Operate		Explanation	
	When does Cyber Stalking happen	60	Explanation	1
	Denial of Service		Definition, Diagrammatic Representation, Explanation	
	Purposes of Hacking	60	Purpose, Explanation	1
	About Hackers, Crackers and Phreaks		Definition, Example with Explanation	
	Online Fraud	60	Explanation with Example	1
	Pronography		Explanation with Example	
UNIT : IV (LECTURE HOURS:8)				
Software Piracy	Examples of Software Piracy	120	End user copies, Hard disk loading, Counterfeiting, downloads from the Inter net	2
	Spoofing	60	Definition, Usenet Newsgroup, Explanation	1
	Virus Dissemination	60	Definition, Explanation.	1
	Typical action of a virus	120	Explanation, Diagrammatic representation of a Virus	2
	World's worst virus attacks	120	Types Explanation-Love letter, Klez, Melissa, Nimda, Anna Kournikova worm, Asset by Threat, Child Pornography, Cyber	2

			Contraband, Cyber Laundering, Cyber Stalking, Cyber terrorism, Cyber theft ,Examples	
UNIT : V (LECTURE HOURS: 7)				
	Types of Cyber Crime	120	Spam, Fraud, Obscene or Offensive Content, Harassment, Drug Trafficking, Cyber Terrorism, Cyber Bullying, Piracy	2
	Preventing Cyber Crimes/Safety Measures	120	General Guidelines on Cyber Safety, Email Safety, Virus Warnings	2
	Protect your Personal Computer	60	Important Guidelines, Explanation	1
	Use a Firewall	120	Software Explanation, Make it a family rule to never give out personal information, Why you should report cyber crime, How to report a Cyber Crime filling a complaint/Writing an application letter	2

TEXT BOOK:

T1. "Information Security And Cyber Laws", Ankur Shree Aggarwal, Prof.Sanjeev Kumar Sharma, AnuradhaTyagi.Shalu Goel,1st Edition , Vayu Education of India, New Delhi.

REFERENCES:

R1. "Internet Privacy Kit", Marcus Goncalves, Que Corporation, 1997.

R2. "Internet Security", OthmarKyas, International Thomson Computer Press, 1997.

SEMESTER V
SKILL BASED COURSE – II
ADVANCED IP SERVICES

OBJECTIVES:

- Enable the students to learn the advanced principles and techniques of IP services.
- To understand the concept of advanced IP Configurations, VLAN, STP, OSPF, EIGRP, PPP WAN, ACLs, MISC LAN.
- To understand the concept of networking and able to configure, verify, and troubleshoot complex computer networks.

PEDAGOGY:

- Teaching aids used are Black board, OHP, Projector, Demonstration and Group Discussion.

TOTAL HOURS/WEEK – 4				
TOTAL HOURS/SEMESTER – 60				
CIA	**CO-CURRICULAR ACTIVITIES**	**LECTURE**	**TUTORIAL**	**TOTAL**
8	**4**	**48**	**-**	**60**

UNIT: I (LECTURE HOURS :10)				
TOPIC(S)	**SUB TOPIC(S)**	**MINUTES**	**KEY POINT(S)**	**HOUR(S)**
	Virtual LAN Concepts	60	Definition, Creating two broadcast domains with Switches and VLANs	1

Virtual LAN Concepts with Trunks	Creating Multiswitch VLANs using Trunking		VLAN Trunking, Tagging and Identifier	
	VLAN Tagging Concepts	60	VLAN Trunk, VLAN Trunking between two Switches	1
	The 802.1Q and ISL VLAN Trunking Protocols	60	802.1Q Trunk, IEEE 802.1Q and Inter-Switch Link	2
	Forwarding Data Between VLANs	60	Campus LAN	
	Routing Packets Between Virtual LAN with a Router	120	Layer 2 Switch Does Not Route between the VLANs	2
	Routing Packets with a Layer 3 Switch		Routing between the Two VLANs using Trunk on the Router	
	Creating VLANs and Assigning Access VLANs to an Interface	60	How to create VLAN , Give the VLAN a name and Assign interfaces to a VLAN	1
	VLAN Trunking Protocol	60	Cisco Protocol, VTP Modes	3
	VLAN Trunking Configuration	60	Type of Trunking, Administrative mode, Switch port mode	
	Implementing Interfaces Connected to Phones	60	Data and Voice VLAN Concepts	
UNIT: II (LECTURE HOURS : 08)				
Spanning Tree Protocol Concepts	Spanning Tree Protocol	60	Two goals of STP	2
	The Need for Spanning Tree		Prevents three common problems in Ethernet LANs, Broadcast Storm	
	What IEEE 802.1D Spanning Tree does	60	Blocks the port to break the loop	
	How Spanning Tree Works	60	Spanning Tree Algorithm, three Criteria	1

	Influencing and Changing the STP Topology		Making configuration changes to influence the STP Topology	
OSPF Concepts and Operation	OSPF Overview	120	Link State protocols builds IP routes	2
	Topology Information and LSAs		LSA and LSDB Relationship	
	Applying Dijkstra SPF Math to Find the Best Routes	60	Applying LSA and calculating routes with three phases	1
Becoming OSPF Neighbors	The Basics of OSPF Neighbors	60	OSPF Neighbors	2
	Meeting Neighbors and Learning their Router ID	60	OSPF Hello Process	
UNIT: III (LECTURE HOURS :10)				
Understanding EIGRP Concepts	Introduction to EIGRP	60	Timeline for IG IGPs	2
	Basic Distance Vector Routing Protocol features	60	DV or LS routing Protocols	
	The concept of a Distance and a Vector	60	Information learned using DV protocols	3
	Full Update Messages and Split Horizon	60	Couple of functions to Update Messages and Split Horizon	
	Route Poisoning	60	Route Poisoning	
EIGRP Operations	EIGRP Neighbors	60	Four settings	2
	Exchanging EIGRP Topology Information	60	EIGRP update Messages	
	Calculating the best routes for the Routing Table	60	Metric calculation	3
	EIGRP Convergence	120	Feasible distance and Reported distance	
UNIT: IV (LECTURE HOURS : 10)				

Access Control Lists Basics	ACL Location and Direction	120	Way to identify the different types of packets	2
	Matching Packets		Location and Direction of ACL	
	Taking Action When a Match Occurs		Two actions	
	Types of IP ACLs	60	ACLs features	1
Standard Numbered ACLs	List Logic with IP ACLs	60	List of one or more configuration commands	1
	Matching Logic and Command Syntax	60	Matching command and Matching Exact IP address, Matching any/all IP Addresses	1
Advanced Access Control Lists	Extended Numbered IP Access Control Lists	60	Modifying larger variety of packet header fields	1
	Matching the Protocol, Source IP and Destination IP	60	IP header, with focus on required fields in extended IP ACLs	1
	Matching TCP and UDP Port Numbers	60	IP header followed by a TCP header and Port Number fields	1
	Named IP Access Lists		Using for filtering Packets	
	Editing ACLs Using Sequence Numbers	60	Applying new style for Editing ACLs	1
	Numbered ACL Configuration Vs Named ACL Configuration	60	Adding and displaying a Numbered ACL Configuration	1
UNIT: V (LECTURE HOURS :10)				
Miscellaneous LAN	Securing Access with IEEE 802.1x	60	LAN is built with cable to run each desk	3
	AAA Authentication	60	How to secure network devices	
	AAA Login Process	60	Install and configure AAA server	
	TACACA+ and RADIUS Protocols	60	Encrypt the passwords	2
	AAA Configuration Examples	60	Configure a Router or Switch	

	DHCP Snooping Basics	60	Acts like a firewall or an ACL	1
	An Example DHCP-based Attack	60	DHCP Attack supplies good IP Address but wrong Default Gateway, DHCP attack leads to Man-in-the-Middle	1
	How DHCP Snooping Works	120	Summary of Rules for DHCP Snooping Configure one switch rather than an act of multiple switch	3
	Improving Design and Availability with Chassis Aggregation	**60**	Key features switch Aggregation	

TEXT BOOK:

CCNA Routing and Switching ICND2 200-105, Official Cert Guide, Wendell Odom, CCIE NO.1624 Ciscopress.com

WEBSITE:

www.digiterati.com

REFERENCE BOOK:

Computer Networks - Andrew S Tanenbaum, Edition: 5th, Prentice Hall.

SEMESTER – VI

MAJOR PAPER

IPV-4 ROUTING

OBJECTIVES:

- Understanding the Concept of IPV-4 Routing.
- Understanding the implementation of BGP,PPP, and HSRP for First-Hop Routing.
- Getting familiarization with Quality of Service in networking &SDN and Network Programmability.

PEDAGOGY:

- Teaching aids used are Black board, OHP, Projector, Demonstration and Group Discussion.

TOTAL HOURS/WEEK-6				
TOTAL HOURS/SEMESTER-90				
CIA	**CO-CURRICULAR ACTIVITIES**	**LECTURE**	**TUTORIAL**	**TOTAL**
12	**6**	**72**	**-**	**90**

UNIT I: (LECTURE HOURS: 16)				
TOPIC(S)	**SUB TOPIC(S)**	**MINUTES**	**KEY POINT(S)**	**HOURS**
Implementing External BGP	BGP Concepts	60	Border Gateway Protocol	1
	Advertising Routes with BGP		Using BGP between Autonomous Systems	
	Internal and External BGP	120	eBGPVsiBGP	2
	Choosing the Best Routes with BGP		ASNs and Shortest AS_Patn as Chosen at ISP3	
	eBGP and the Internet Edge	60	Connection between ISP	1
	Internet Edge Designs and Terminology		Single homed, Dual homed, Single Multihomed, Dual Multihomed	
	Advertising the Enterprise Public Prefix into the Internet	60	ISP Propagates	1
	Learning Default Routes from the ISP	60	ISP Advertises a Default Route; IGP Propagates	1
	eBGP Configuration and Verification	60	Enabling routing protocol in interfaces	1
	BGP Configuration Concepts	60	Steps for Advertising a Public prefix from the Enterprise to an ISP	1

	Configuring eBGP Neighbors Using Link Addresses	60	Using Link Addresses for BGP Peering in a Single Homes design	1
	Verifying BGP Neighbors	60	eBGP neighbor states and TCP connection	1
	Administratively Disabling Neighbors	60	Neighbour state with the Neighbor Shut Down	1
	Injecting BGP Table Entries with the network command	60	BGP Network	1
	Injecting Routes for a Classful Network	60	All of Class C network used as DMZ Subnet	1
	Advertising Subnets to the ISP	60	Using Two Subnets of DMZ and NAT	1
	Advertising a Single Prefix with a Static Discard Route	120	BGP Configuration with Discard Route	2
	Learning a Default Route from the ISP		Receiving a Default Route for Router ISP1	
UNIT II: (LECTURE HOURS: 12)				
TOPIC(S)	**SUB TOPIC(S)**	**MINUTES**	**KEY POINT(S)**	**HOURS**
Implementing PPP Concepts	PPP Framing	120	PPP Framing	2
	PPP Control Protocols		Defines the set of layer 2 protocols	
	PPP Authentication	60	Authentication of devices	1
	Implementing PPP		Configuring PPP	
	Implementing PPP CHAP	60	Configuring PPPCHAP	3
	Implementing PPP PAP	120	Configuring PPP PAP	
	Implementing Multilink PPP	120	Configuring Multilink PPP	2
	Multilink PPP Concepts	60	Multilink PPP	1
	Configuring MLPPP	180	Configuring ML PPP	3
	Verifying MLPPP		Verify ML PPP	
UNIT III: (LECTURE HOURS: 14)				
Quality of Service (QoS)	Introduction to QoS	180	WAN interfaces and LAN interfaces	3
	QoS: Managing Bandwidth, Delay, Jitter and Loss		Bandwidth, Delay, Jitter and Loss	

	Types of Traffic	120	QoS Plan	2
	Data Applications		Interactive Data Application	
	Voice and Video Applications	180	Creating VoIP Packets with an IP Phone	3
	QoS as Mentioned in		QoS tools	
	QoS on Switches and Routers		Packet and Frame	
	Classification and Marking		Classifies packets	
	Classification Basics	180	Classification for Queuing in a Router	3
	Matching (Classification) Basics		Systematic classification and Marking for the Enterprise	
	Classification on Routers with ACLs and NBAR	180	Classification with Five Fields used by extended ACLs	3
	Marking IP DSCP and Ethernet CoS		Creating classes of traffic	
UNIT-IV (LECTURE HOURS: 14)				
Quality of Service (QoS) with Congestion, Shaping and Policing	Congestion Management	120	LIQ always Schedule Voice Packet Next	2
	Round Robin Scheduling (Prioritization)	180	Data, Voice and Video	3
	Low Latency Queuing		Qos Tools	
	A Prioritization Strategy for Data, Voice and Video	240	Effect of a Policer and Shaper on an offered Traffic load	4
	Shaping and Policing		Ethernet WAN: Link Speed Vs CIR	
	Policing		Scheduling with LLQ and CBWFQ	
	Where to Use Policing	240	Shaping Time Interval	4
	Shaping		To reduce packet loss	
	Setting a Good Shaping Time Interval for Voice and Video		Flow control mechanism	
	Congestion Avoidance	60	Mechanism of Congestion Avoidance	1
UNIT V: (LECTURE HOURS: 16)				
TOPIC(S)	**SUB TOPIC(S)**	**MINUTES**	**KEYPOINTS**	**HOURS**
Implementing HSRP for First-Hop Routing	FHRP and HSRP Concepts	180	HSRP, VRRP, GLBP	3
	The Need for		All traffic goes to R1 ,R2 is	

	Redundancy in Networks		stand by	
	The Need for a First Hop Redundancy Protocol		Packet sent through R2 once it takes over for failed R1	
	The Three Solutions for First-Hop Redundancy	60	Load balancing with HSRP by using different active router per subnet	1
	HSRP Concepts		Show commands	
	HSRP Failover	60	HSRP configuration on R1 and R2	1
	HSRP Load Balancing		No Preemption keeps R1 as standby after R1 recovers	
	Implementing HSRP	180	Versions	3
	Configuring and Verifying Basic HSRP		Commands	
	HSRP Active Role with Priority and Preemption		Finding HSRP Configuration in the show standby command output	
	HSRP Versions		HSRP Misconfiguration Scenarios and Expected results.	
SDN and Network Programmability	SDN and Network Programmability Basics	180	Controls the Data plane	3
	The Data, Control and Management Planes		Overhead work	
	The Data Plane		Switch Data Plane	
	The Control Plane		Controller	
	The Management Plane		Centralized control plane and a distributed data plane	
	Cisco Switch Data Plane Internals	60	Southbound Interface	1
	Controllers and Network Architecture		Northbound Interface	
	Controllers and Centralized Control	60	Opening Networking Foundation	1
	The Southbound Interface		Wide variety of SDN	
	The Northbound Interface	60	SDN for different parts of network	1
	Open SDN and Openflow		Controlling the ACI Data Center Network using the APIC	

TEXT BOOK:

CCNA Routing and Switching ICND2 200-105, Official Cert Guide, Wendell Odom, CCIE NO.1624 Ciscopress.com

WEBSITE: www.digiterati.com

SEMESTER-VI

MAJOR PAPER

ANDROID PROGRAMMING

OBJECTIVES:

- Understand how Android applications work, their life cycle, manifest, intents, and how to use external resources to design and develop useful Android applications.
- Design and develop useful Android applications with compelling user interfaces by using, extending, and creating our own layouts, menus and views.

PEDAGOGY:

- Teaching aids used are Black board, OHP, Projector, Demonstration and Group Discussion.

TOTAL HOURS/WEEK - 6			
TOTAL HOURS/SEMESTER - 90			
CIA	**CO-CURRICULAR ACTIVITIES**	**LECTURE**	**TUTORIAL**
12	**6**	**72**	**-**

UNIT : I (LECTURE HOURS: 14)				
TOPIC(S)	**SUB TOPIC(S)**	**MINUTES**	**KEY POINT(S)**	**HOURS**
Android overview	Types of devices	60	Phone, Tablets, Readers, Cameras, Home automation systems, Home appliances, Vehicle systems, Game consoles.	2
	Types of Apps		Category, functionality of apps.	
	A brief history		Open Handset Alliance, Android Open Source Project (AOSP), Software Development Kit(SDK).	
	Versions		Android versions, Code name, API, distribution and description.	
	System Architecture	60	Android stack, Dalvik virtual machine (VM), Java virtual machine (JVM).	
	How Apps are compiled and run		Android system architecture, Integrated Development	

			Environment (IDE), Android Virtual Device (AVD), Android Debug Bridge (ADB).	
TOPIC(S)	**SUB TOPIC(S)**	**MINUTES**	**KEY POINT(S)**	**HOURS**
A simple Android app	The user interface	60	The tip calculator app with the Soft keyboard displayed, Description.	3
	The XML for the user interface		The activity_tip_calculator.xml file.	
	The XML for the display text	60	The activity_tip_calculator.xml file, The strings.xml file.	
	The java source code	60	The TipCalculatorActivity.java file.	
	The Android Manifest		The Androidmanifest.xml file.	
How to work with existing projects	An introduction to eclipse projects	60	The Eclipse workbench with a project.	6
	How to set the workspace		The workspace Launcher dialog box, How to switch the Eclipse workspace.	
	How to import a project in to the workspace	60	The dialog box for importing an existing project, How to import a project.	
	How to remove a project from the workspace		The dialog box for removing a project from the work space.	
	How to work with the user interface	60	The Graphical layout editor, The XML editor, The syntax for an xml comment.	
	How to work with other XML resources	60	The strings.xml file.	
	How to work with the java code	60	The java code for the Tip Calculator app.	
	How to set the run configuration		The Run Configurations dialog box.	
	How to run an app on a physical device	60	Package Explorer, Eclipse.	

	How to run an app on an emulator		The android device chooser dialog box.	
TOPIC(S)	**SUB TOPIC(S)**	**MINUTES**	**KEY POINT(S)**	**HOURS**
How to work with a new project	How to create a new project	60	The new Android application dialog box.	2
	How to work with the starting code		The code editor with the starting xml code for an activity, The code editor with the starting java code for an activity.	
	How to use the code completion feature	60	The code editor with a code completion list.	
	How to detect and correct errors and warnings		The code editor with an error displayed and a possible solution.	
The Tester app	The user interface	60	Tester app running in an emulator.	1
	The XML for the user interface		The activity_test.xml file.	
	The java source code		The TestActivity.java file.	
UNIT : II (LECTURE HOURS: 14)				
TOPIC(S)	**SUB TOPIC(S)**	**MINUTES**	**KEY POINT(S)**	**HOURS**
How to develop the user interface	The Tip Calculator app	60	The use interface for the Tip Calculator.	4
	How to work with a layout		The default layout for an activity in a new project.	
	How to add widgets to a layout	60	A layout after some widgets have been added to it.	
	How to set the display text		A layout after the text has been set for the widgets, The Resource Chooser dialog box.	
	How to work with the strings.xml file	60	The strings.xml file for the Tip Calculator app.	
	How to set properties		A layout after some properties of the widgets have been set.	

	Common properties	60	Common properties for layouts, Common properties for widgets, predefined values for setting height and width, common units of measurement.	
	The XML for the user interface		The XML for the user interface.	
TOPIC(S)	**SUB TOPIC(S)**	**MINUTES**	**KEY POINT(S)**	**HOURS**
How to write the java code	How to work with an Activity	60	The default java code for an activity.	5
	How to get references to widgets		An activity that gets references to the widgets.	
	How to handle the EditorAction Event	60	An activity that handles the EditorAction event, A few constants from the Editorinfo class.	
	How to get and set text for widgets		getText method, setText method.	
	How to handle the Click event	60	Import the interface for the listener, Implement the interface for the listener, Set the listener.	
	The life cycle of an activity		Resumed, paused, stopped, created state, started state.	
	How to save and restore values	60	How to import the SharedPreferences class and Editor class, How to set up the instance variable, How to use onPause method to save values, How to use onResume method to restore values.	
	How to use the documentation for the android API		The documentation for the Activity class.	
	The java code for the APP	60	Java code.	
Basic skills for testing and debugging	Typical test phases	60	The Tip Calculator with logical error, Four test phases.	1
	How to check the layout		Graphical Layout editor for the Android 2.2 platform.	

	The three types of errors		Syntax errors, runtime errors, exceptions, logic errors.	
	How to handle runtime errors		The error that's displayed when an app crashes, The LogCat view after a crash.	
TOPIC(S)	**SUB TOPIC(S)**	**MINUTES**	**KEY POINT(S)**	**HOURS**
How to trace code execution	How to use LogCat logging	60	Eclipse with the LogCat view displayed, A few methods of the Log class, How to use the Log class.	1
	How to use toasts		Two methods of the Toast class, Two constants of the Toast class, How to display a toast.	
How to use the debugger	How to set and remove breakpoints	60	Java perspective with a breakpoint.	2
	How to step through code		Step through, Step Into button.	
	How to inspect variables		This, scope.	
	How to inspect the stack trace	60	The Debug perspective.	
	How to configure step filters		The Step Filtering preferences, Common packages to add to step filtering.	
How to configure your emulators	How to add an emulator for an old phone	60	The device definition for a phone with a hard keyboard and Dpad, How to create a new device definition, How to create an emulator.	1
	How to work with an emulator for an old phone		The soft keyboard on an emulator for an old phone.	
	How to add an emulator for a tablet		An emulator for a tablet.	
UNIT : III (LECTURE HOURS: 15)				

An introduction to layouts and widgets	A summary of layouts	60	RelativeLayout, LinearLayout, Table Layout, FrameLayout, AbsoluteLayout, GridLayout.	1
	A summary of widgets		TextView, EditText, Button, CheckBox.	
	The view hierarchy		ViewGroup, TextView, ProgressBar, ImageView.	
TOPIC(S)	**SUB TOPIC(S)**	**MINUTES**	**KEY POINT(S)**	**HOURS**
How to work with layouts	How to use a linear layout	60	A linear layout with vertical orientation and two buttons, Common attributes for working with linear layouts.	2
	How to use a table layout		A table layout with two rows and four columns, Common attributes for working with table layouts.	
	How to use a frame layout	60	A frame layout that displays an image behind some text.	
	How to nest layout		Nested linear layouts.	
	How to provide a landscape layout		The location of the xml files, the xml for landscape orientation.	
How to work with widgets	How to use editable text views	60	The soft keyboard for an editable text view for an email address, Two attributes of an EditText widget.	4
	How to use check boxes		A common xml attribute for check boxes, Two common java methods for check boxes.	
	How to use radio buttons	60	Three radio buttons in a radio group with vertical orientation, Three radio buttons in a radio group with horizontal orientation.	
	How to use spinners		The xml code, the array in the strings.xml file, Common methods for spinners.	
	How to use seek bars	60	Two common XML attributes for seek bars, Two common java methods for seek bars.	

	How to display images		Two attributes of an ImageView widget, Four qualifiers for the drawable folder.	
	How to show and hide widgets	60	A layout with three rows, The same layout with the per person amount hidden, A method of the view class.	
	How to add scroll bars		A scrollable layout.	
TOPIC(S)	**SUB TOPIC(S)**	**MINUTES**	**KEY POINT(S)**	**HOURS**
A summary of listeners	High-level events	60	Listeners for high-level events.	1
	Low-level events		Listeners for low-level events.	
Four techniques for handling events	How to use the current class as the listener	60	Use the current class as the listener.	2
	How to use a named class as the listener		Import the interface for the listener, Use the current class as the listener, Use a separate named class as the listener.	
	How to use an anonymous class as the listener		Use an anonymous class as the listener.	
	How to use an inner anonymous class as the listener	60	Use an inner anonymous class as the listener.	
	When to use each technique		Consistency.	
How to handle high-level events	How to handle events for check boxes and radio buttons	60	A checkbox, An event handler for a check box, Another event handler for a check box, A method of the view class.	2
	How to handle events for radio groups		Three radio buttons in a group, An event handler for a radio group, Another event handler for a radio group.	

	How to handle events for spinners	60	A spinner, An event handler for a spinner.	
	How to handle events for seek bars		A seek bar and a label, An event handler for a seek bar.	
TOPIC(S)	**SUB TOPIC(S)**	**MINUTES**	**KEY POINT(S)**	**HOURS**
How to handle low-level events	How to handle Key events	60	An event handler for the key event, some constants from the KeyEvent class.	1
	How to handle Touch events		An event handler for a Touch event, some constants of the MotionEvent class, some methods of the MotionEvent class.	
The Tip calculator app	The user interface	60	The user interface.	2
	The java code for the activity	60	The java code for the activity.	
UNIT : IV (LECTURE HOURS: 15)				
An introduction to themes and styles	Three themes	60	Theme. Light, Theme.Holo.Light, Theme.Holo.Light.DarkActionBar.	1
	The theme framework that's generated by Eclipse		The styles.xml file in the res\values directory.	
How to work with styles	How to define a style	60	A style that overrides one property, A style that inherits a user defined style, A style that inherits a user defined style and overrides two properties , Another way to code the previous style , How to inherit multiple user defined styles.	1
	How to apply a style		The reference chooser dialog for a style.	

	How to create a style sheet		A styles.xml file in the res\values directory with four user defined styles.	
TOPIC(S)	**SUB TOPIC(S)**	**MINUTES**	**KEY POINT(S)**	**HOURS**
How to work with themes	How to modify a theme	60	A styles.xml file in the res\values directory that customizes a theme.	2
	How to modify the text appearance for a theme		Some built in styles for controlling text appearance.	
	A summary of built-in themes	60	Some built in themes.	
	How to apply themes		Theme.Holo.Light.Dialog, An AndroidManifest.xml file that uses a built-in theme.	
How to work with colors	How to define colors	60	A colors.xml file in the res\values directory.	1
	How to apply colors		How to apply colors to a widget, How to apply colors to a style, How to apply colors to a theme.	
How to work with menus	An introduction to menus	60	An activity with an options menu that has two items, The same options menu displayed from an action overflow icon.	2
	How to define a menu		The file that contains the XML for the menu, The XML for the menu, some attributes of a menu item.	
	How to display options menu	60	onCreateOptionsMenu method, The code that displays the menu.	
	How to handle option menu events		The code that handles the menu item events.	
	How to start a new activity		Code that starts a new activity, Code that uses menu items to start new activities.	

TOPIC(S)	SUB TOPIC(S)	MINUTES	KEY POINT(S)	HOURS
How to work with preferences	An introduction to preferences	60	The settings activity, The dialog for the rounding item.	2
	How to define preferences		The file that contains the xml for the preferences, The xml for the preferences, Some attributes that apply to all preference items, Some attributes that apply to List preference.	
	How to display preferences in an activity		The SettingsActivity class.	
	How to display preferences in a fragment	60	The SettingsFragement class, The SettingsActivity class.	
	How to get preferences		Define the instance variables for the preferences, Set the default values in the preferences file, Get the SharedPreferences object, Get the preferences, Some get methods of the SharedPreferences object.	
	How to use preferences		Use the “Remember Tip percent” preference in the onResume method, Use the “Rounding” preference in the calculateAndDisplay method.	
More skills for working with preferences	How to group preferences	60	Setting activity that uses categories, The xml for the preferences.	1
	How to enable and disable preferences		A settings activity that uses dependencies, The xml for the preferences, The dependency attribute.	
	How to use java to work with preferences		A class that works with preferences.	

TOPIC(S)	SUB TOPIC(S)		KEY POINT(S)	HOURS
An introduction to fragments	Single-pane and multi-pane layouts	60	Two activities displaying two fragments, One activity displaying two fragments in both landscape and portrait orientation.	1
	How to use support libraries		A JAR file for the support library for API 4 and higher , How to install the Android Support library, The SDK directory that contains the Android support library, How to make the support library available to a project.	
	The life cycle methods of a fragment		The lifecycle methods of a fragment- OnResume(),onStart (), onActivityCreate(),onCreateView(), onCreate(), onAttach(), onPause(), onStop().	
How to use single-pane layouts for small screens	How to create the layout for a fragment	60	The layout for a fragment.	2
	How to create the class for a fragment		Method of the Fragment class, The declaration for the TipCalculatorFragment class, The onCreate method, The onCreateView method.	
	How to display a fragment in an activity	60	The activity_main.xml file, The TipCalculatorActivity class.	
	How to create a preference fragment		The SettingsFragment class.	
	How to display a preference fragment in an activity		The activity_setting.xml file, the SettingsActivity class.	
TOPIC(S)	**SUB TOPIC(S)**	**MINUTES**	**KEY POINT(S)**	**HOURS**

How to use multi-pane layouts for large screens	How to add multiple fragments to a layout	60	Two layout files for the main activity.	1
	How to detect large screens		Two qualifiers, The layout files for devices with large screens.	
	How to detect screen width		Smallest width qualifiers examples, The layout files for devices with a minimum screen size.	
	How to control the soft keyboard		Some values for the imeOptions attribute, Attributes of an EditText widget that can control the soft keyboard.	
Other skills for working withfragments	How to get a reference to a fragment	60	Method of the Activity and Fragment classes, Method of the FragmentManager class.	1
	How to replace one fragment with other		Method of the FragmentManager class, Method of the FragmentTransaction class.	
UNIT : V (LECTURE HOURS: 14)				
TOPIC(S)	**SUB TOPIC(S)**	**MINUTES**	**KEY POINT(S)**	**HOURS**
An introduction to databases	The user interface for the Task List APP	60	The Task List activity, the Add/Edit activity.	2
	An introduction to SQLite		SQLite, Data types supported by SQLite, The location of the SQLite database file for the TaskList app.	
	An introduction to the Task List database	60	The diagram for the Task List database, The SQL statement that creates the LIST table, The SQL statement that creates the Task table, The SQL statement that drops the List table and Task table.	
	The business objects for the Task List app		The List class, The Task class.	
TOPIC(S)	**SUB TOPIC(S)**	**MINUTES**	**KEY POINT(S)**	**HOURS**

How to create a database class	How to define the constants for a database	60	The TaskListDB class.	2
	How to define the SQL statements that create a database		The TaskListDB class	
	How to create or upgrade a database	60	The TaskListDB class , A method of the SQLiteDatabase class	
	How to open and close a database connection		The TaskListDB class, Two methods of the SQLiteOpenHelper class, A method of the SQLiteDatabase class.	
How to add public methods to a database class	How to retrieve multiple rows from a table	60	The TaskListDB class, A method of the SQLiteDatabase class, Some methods of the Cursor class.	2
	How to retrieve a single row from a table		The TaskListDB class, Some methods of the Cursor class.	
	How to get data from a cursor	60	The TaskListDB class, More methods of the Cursor class.	
	How to insert, update and delete rows		The TaskListDB class, Two methods of the SQLiteDatabase class, One method of the ContentValues class, Another method of the SQLiteDatabase class.	
How to test the database class and clear its data	How to test the database class	60	Code that tests the database class, Sample text displayed on the user interface.	2
	How to clear test data from a device		The settings app displaying info about the Task List app.	
TOPIC(S)	**SUB TOPIC(S)**	**MINUTES**	**KEY POINT(S)**	**HOURS**

How to test the database class and clear its data	How to use the DDMS perspective to work with database files	60	The database files, How to view a database.	
	How to use the SQLite database browser		The SQLite database browser, The URL for downloading the SQLite database browser.	
How to use tabs	How to add the TabManager class to your project	60	A tab widget displayed in the Graphical Layout editor, The xml for the layout.	2
	The layout for an activity that displays tabs		A tab widget displayed in the Graphical Layout editor, The xml for the layout.	
	The class for an activity that displays tabs	60	The class for an activity that displays tabs.	
	The class for a fragment that displays tab content		The class for a fragment that displays tab content.	
How to use a custom adapter	A layout for a list view item	60	The listview_task layout, The xml for the layout.	2
	A class that extends the layout for a list view item		A class that extends the layout.	
	A class for a custom adapter	60	A class for a custom adapter.	
	A class for a fragment that uses a custom adapter		A class for a fragment that uses a custom adapter.	
TOPIC(S)	**SUB TOPIC(S)**	**MINUTES**	**KEY POINT(S)**	**HOURS**
The Task List app	The user interface	60	The Task List activity, the Add/Edit activity.	2
	The activity_task_list		The activity_task_list menu.	

	menu			
	The TaskListActivity class		The TaskListActivity class.	
	The activity_add_edit and spinner_list layout	60	The activity_add_edit layout, The spinner_list layout.	
	The activity_add_edit menu		The activity_add_edit menu.	
	The AddEditActivity class		The AddEditActivity class.	

TEXT BOOKS:

T1. Training and Reference Murach's Android programming by Joel Murach, Mike Murach&Associates (2013).

. Chapters 1,2,3,4,5,6,7,8,9,13,14

REFERENCES:

R1. Android Apps for Absolute Beginners by Wallace Jackson, Second Edition, Apress (2013).

SEMESTER – VI

MAJOR PRACTICAL

PROGRAMMING LAB-VI (MONGO DB & IPV-4 ROUTING)

OBJECTIVES of IPV-4 ROUTING:

- Understanding the Concept of IPV-4 Routing.
- Understanding the implementation of BGP, PPP, and HSRP for First-Hop Routing.
- Getting familiarization with Quality of Service in networking &SDN and Network Programmability.

TOTAL HOURS/WEEK-6				
TOTAL HOURS/SEMESTER-90				
CIA	**CO-CURRICULAR ACTIVITIES**	**LECTURE**	**TUTORIAL**	**TOTAL**
12	**6**	**72**	**-**	**90**

(UNIT-I, UNIT-II, UNIT-III--MONGO DB Practical Programs)- 36 Hours
(UNIT-IV & UNIT-V-- IPV-4 ROUTING Practical Programs)- 36 Hours

UNIT-I

1. Create a Student Database in MongoDB using "use" Command.
2. Create an Employee Database in MongoDB using "use" Command.
3. Create a collection for student Database in MongoDB. (Consider the "db.collectionName" pattern)
4. Create a collection for Employee Database in MongoDB. (Consider the "db.collectionName" pattern)
5. Run a server with the following instruction.
   i. Open command prompt as administrator
   ii. Navigate to MongoDB root
   iii. Create a folder "data"
   iv. Run "mongod –dbpath ../data"

6. Use show command and list all database in the MongoDB.
7. Use show command and list all collection in the Mongo shell.
8. Customize prompt to Display number of operations in the Mongo Shell.
9. Customize prompt to Display database and hostname in the Mongo Shell.

UNIT-II

10. Insert a single document into an inventory collection. The inventory collection has the following fields: item, quantity, tags and size.
11. Insert a multiple documents into an inventory collection. The inventory collection has the following fields: item, quantity, tags and size.
12. Use find () method for the following statement
    i. Select all documentsin an inventory collection
    ii. Select all documents in an inventory collection and "status is D".
    iii. Select documents in an inventory collection and apply "in" condition for status "A" and "D".
    iv. Select documents in an inventory collection and apply "and as well as or" condition.

13. Use update() command for the following statement
    i. Apply updateone command and update a single document for inventory collection (item=paper, size.uom=cm and status=p)
    ii. Apply updatemany command and update a multiple document for inventory collection(qty is less than 50, size.ucom=in and status=p)

14. Use deleteMany command and delete all documents from the inventory collection.

15. Remove only one document that matches a “status is D” in inventory collection.

UNIT-III

16. Create text index for stores collection and fields are name=text , description=text.
17. Use $match and $group operator and calculate the total views for articles that contains a word (cake) in article collection.
18. Use $sort and $meta operator and return results sorted by text search score for article Collection.
19. Create an ascending index on a single field and create an index on embedded document for record collection.
20. Create the replica set in the mongo shell and test the configuration.

UNIT-IV –(24 Hours)

Program 1:Configure eBGP. (8 Hours) Configure eBGPrelation between two cisco1941 or 1841 router to route traffic from an enterprise network towards ISP. Make sure that the resource network (which is connected off ISP side) is also advertised properly.

Program 2: Configure lease line. (8 Hours)

Configurelease line between two ISRs’ by simulating PPP activity using LCP and NCP.

Program 3: Configure a Multi-switched based VLAN network. (8 Hours)

Configurea Multi-switched based VLAN network using catalyst family switches and make those switches to work on appropriate trunking with VTP modes. Ensure that all the VLAN databases maintain an equal revision number.

UNIT-V –(12 Hours)

Program 4: Deploy STP in multi-switched networks. (6 Hours)

Deploy STP in multi-switched networks of catalyst switches to ensure the availability of individual root bridges per VLANs. Use three VLANs.

Program 5: Routing with Djikstra’s algorithm using LB catalyst switches. (6 Hours)

Perform and demonstrate routing with Djikstra’s algorithm using LB catalyst switches to work like routers and demonstrate the shortest path algorithm manually.

*****

SEMESTER-VI

ELECTIVE-III
MONGODB

Objectives

To enable the learners to

- Understand the purpose of NOSQL database
- Familiarize the concepts of MongoDB

Pedagogy

- Using lecture, group discussion, LCD, Demonstration and seminars

TOTAL HOURS/WEEK-4+1				
TOTAL HOURS/SEMESTER – 75				
CIA	**CO-CURRICULAR ACTIVITIES**	**LECTURE**	**TUTORIAL**	**TOTAL**
10	**5**	**48**	**12**	**60**

UNIT : I (LECTURE HOURS:9 TUTORIAL HOURS:3)

TOPIC(S)	SUB TOPIC(S)	MINUTES/ SUB TOPICS	KEY POINT(S)	HOURS/ TOPICS
Introduction to MongoDB	What is MongoDB	60	Database, collection, Document	1
	Document oriented database		Document	
	Features of MongoDB		Ad hoc queries, indexing, replication, duplication of data, load balancing, map reduce, schema less, java script, high performance	
	NoSql Databases	60	NoSql Database, Advantages	1
	Common terms in MongoDB		_id, collection, cursor, Database, Document, Field, JSON	
	Difference between MongoDB& RDBMS, Sql&Nosql	60	Table, row, column, joins Sql, Nosql	1
	MongoDBDatatypes	60	String, integer, Boolean, Double, Min/Max keys, Arrays, object, Null, Symbol, Date	1
Databases and collections	views	60	Create view, view creation, shared view, drop a view, modify a view, supported operation	1
	Capped collections	60	Overview, behavior, restrictions and recommendations, procedures	1
Documents	structure	60	Field name, field value limit	1
	Dot notation		Arrays, embedded documents	

	Document limitation	60	Document size limit, field order, _id field	1
	Other uses of the document structure	60	Query filter documents, update specification documents, index specification documents	1
Tutorial Hour	Create a Student Database using "use" Command	60	Use command	1
	Create an Employee Database using "use" Command	60	Use command	1
	Create a collection for student and Employee	60	Collection name	1
UNIT : II (LECTURE HOURS: 9 TUTORIAL HOURS:3)				
MongoDB Architecture	The Nexus Architecture	120	**Expressive query language & secondary Indexes, Strong consistency, Enterprise Management and Integrations, Flexible Data Model, Scalability and Performance, Always-On Global Deployments**	2
Install MongoDB	Installing MongoDB on windows	120	**MongoDB package, Create a data directory, Execute MongoDB**	2
The mongo shell	Configure the mongo shell	60	Customize the prompt, use an external editor in the mongo shell, change the mongo shell batch size	1
	Access the mongo shell help	60	Command line help, shell help, database help, collection help, cursor help, wrapper object help	1
	Write scripts for the mongo shell	120	Opening the new connections, difference between interactive and scripted mango, scripting	**2**
	Data types in the mongo shell	60	Types, check types in the mango shell	**1**
Tutorial Hour	Install MongoDB	60	Installation	**1**
	un MongoDB server on windows	120	Mongod.exe, mongo.exe	**2**
UNIT : III (LECTURE HOURS: 10 TUTORIAL HOURS:2)				
MongoDB CRUD operation	Insert Document	60	Insert methods	1
	Query Document	120	Nested documents, query on array, embedded documents, project fields to return from query, null or missing fields, iterate a cursor in a mongo shell	2
	Update Documents	60	Update methods	1
	Delete Documents	60	Delete methods	1
Bulk write operations	overview	60	Db.collection.bulk write()	3
	Ordered vs unordered	120	Ordered bulk write, unordered bulk write()	

	operations			
	Bulkwrite method()		Insertone(), updateone(), updatemany(), replaceone(), deleteone(), deletemany()	
SQL to MongoDB Mapping chart	Terminology and concepts	60	Terms and concepts of SQL and MongoDB	2
	Executables		Executables	
	Examples	60	Examples	
Tutorial Hour	Insert Methods	60	Insertone(),insertmany(),insert()	1
	Update methods	60	Updateone(),updatemany(), replaceone(), update()	1
UNIT : IV (LECTURE HOURS: 10TUTORIAL HOURS:2)				
Text search	Text indexes	60	Text indexes	1
	Text search operators	60	Query frame work, aggregation frame work	1
	Text search in the aggregation pipeline	120	Restrictions, text score, calculate the total views for articles that contains a word, return results sorted by text search score, match on text score, specify a language for text search	2
Indexes	Single field indexes	120	Create an ascending index on a single field, create an index on an embedded field, create an index on embedded document	2
	Compound indexes	60	Multi key indexes, text indexes	1
	Hashed indexes	60	Index properties, index bulk operations	1
	Index intersection	60	Index prefix intersection, index intersection and compound indexes, index intersection and sort	1
	Manage indexes	60	View existing indexes, remove indexes, modify an index, rebuild indexes	1
Tutorial Hour	Measure index use	120	Indexing strategies	2
UNIT : V(LECTURE HOURS:10TUTORIAL HOURS:2)				
Replication	Replica set primary	60	Replica set secondary members	1
	Replica set arbiter	120	Example, security	2
Sharding	Shared cluster	60	Shared cluster	3
	Shared keys	60	Shared keys	
	chunks	60	Chunks	
	Advantages of sharding	60	Read/write, storage capacity, high availability	1
	Shared and non-shared collections	120	Shared, non-shared collections	2
	Connecting to a shared cluster	60	Mongos, shared cluster	1
Tutorial Hour	Choosing a shard key	120	Restrictions, collection size, shard key cardinality, shard key frequency, monotonically changing shard keys	2

Text Book:

1.https://docs.mongodb.com/manual/introduction/
2. Kristina chodorow “MongoDB, The definitive guide”, 2nd Edition, O’Reilly Publishing, 2013.
3. Shashank Tiwari “Professional NoSQL”, 1st edition, Wiley India Pvt Ltd, 2015.

References:

1.Provider Name: MongoDB University

Course Name: MongoDB for Developers
University: MongoDB University
Instructor: Andrew Erlichson
Course Link: https://university.mongodb.com/courses/M101P/about

2. Provider Name: edX.org

Course Name: Introduction to MongoDB using the MEAN Stack
University: MongoDB University
Instructor: ValeriKarpov

Course Link: https://www.edx.org/course/introduction-mongodb-using-mean-stack-mongodbx-m101x-0

SEMESTER VI
MAJOR PAPER
ELECTIVE-III

Enterprise Java Beans

OBJECTIVES:

To enable the students to

Get a standard overview of EJB rationale and its architecture.
Create session beans and entity beans and know to deploy the application in a server.
Use of JDBC to access a SQL database

PEDAGOGY:

Teaching aids used are Black board, OHP, Projector, Demonstration, Analogy and Group Discussion.

TOTAL HOURS/SEMESTER –90		
CIA	**CO-CURRICULAR ACTIVITIES**	**LECTURE**
12	**6**	**72**

UNIT : I (LECTURE HOURS: 14)

TOPIC(S)	SUB TOPIC(S)	REFERENCE & PG. NO.	KEY POINT(S)	MINUTES
The motivation for EJB , Divide and Conquer and Component Architectures	Distributed System	T1: Pg.No. 3 - 12	Standard multitier deployment, role of Application server	180
	Components		Pricing Components, Examples of Different vendors	
	Building large systems		Things to consider Like RMI, Load balancing	
	Application Server		Set of interfaces between components and application server	
Introducing Enterprise JavaBeans , The EJB Ecosystem	Importance of Java	T1: Pg.No. 13-21	3 main reasons and its features	180
	EJB as a Business Solution		Tasks of EJB, GUIs, EJB as back-end to web services	
	The bean Provider, The application Assembler,		Vendors of EJB, Tasks of Application assembler	
	The EJB Deployer, The system administrator		Challenges of EJB deployment, JMX, Responsibilities	
	The container and Server Provider, The tool vendors, Roles in EJB		Runtime environment, IDEs, Parities of EJB and their requirements	
J2EE , Enterprise Beans, distributed objects and Distributed objects middleware	The J2EE technologies	T1: Pg.No.22-37	Java platforms, EEd, EJB, RMI, JNDI, JDBC	180
	Types of Beans		Session bean , Entity bean , Message driven definition, Clients interaction	
	The foundation for EJB		Local/Remote transparency, Tchnologies	
	Explicit Middleware		Gained through APIs, Pseudo code, downsides of this approach	
	Implicit Middleware		Gained through declarations, Tasks and values of this approach	
Constitutes an Enterprise Bean	Enterprise Bean class	T1: Pg.No. 37-44	Interface for all classes (Source code)	120
	EJB Object		Services, Steps to interact with EJB container, Remote interface	
Constitutes an Enterprise Bean	The Home object	T1: Pg.No. 44-54	Responsibilities of Home object, Home interface, Its source code	180
	Local interfaces		Steps involved in creation, Its process, Preview of javax.ejb. EJBHome	

	Deployment Descriptors, Vendor Specific files		Requirements for deployment, Creating ejb-jar files, Terms associated.	
UNIT : II(LECTURE HOURS:14)				
Develop an EJB component	Remote interface	T1: Pg.No. 55-65	Order of operations, Simple java program	180
	Local interface		Source code for implementation, HelloLocal.java	
	Home interface		Methods in Home, Steps to create, Java Source code	
	Local Home interface Bean class		Differences, Exceptions, Create(), Remove(), EJB Contexts,	
Deploying a bean , Understanding how to call beans	The optional EJB client JAR file	T1: Pg.No. 66-74	Steps for deployment, Vendor specific files, Ejb.jar file,	180
	Looking up a home project		RMI-IIOP, Use JNDI, Hello client source code	
Running the system , Introduction to session beans	Server side output & Client side output	T1: Pg.No. 75-84	Methods to invoke, JNDI requirement, Running the client.java	180
	Implementing component interfaces		Reason for component interface, Parameter passing	
	Session bean lifetime		Persistent objects, Non persistent objects,	
	Session bean subtypes		Stateful session bean, Stateless session bean, Its invocation, Method associated	
Special characteristics of stateful session beans	Achieving the effect of Pooling with stateful beans	T1: Pg.No. 84-89	Context Switching, Passivation, Virtual memory, Transaction	120
	The rules governing conversational state		Object serialization, Code of session bean	
	Activation and Passivation Callbacks		Description of methods ejbActivate(), ejbPassivate(), Activation, Passivation process	
Method implementation	A simple stateful session bean	T1: Pg.No. 89-103	Count bean, Its required interfaces, Source code of interfaces, Client's code , I/O process	180
	Lifecycle diagrams for session bean.		Steps involved, Methods required, diagrammatic representation.	
UNIT : III(LECTURE HOURS:14)				
Entity beans - Persistence concepts, Entity beans	Java object serialization	T1: Pg.No. 105-112	RMI, JNDI, Querying objects, Example	120
	Object relational mapping		Rational database, Example account database, Relationship to other data	
	Object databases		ODBMS, Object query language, Object database technology , Vendors	
	About the files that make up an Entity bean		Application , Persistent Logic component, bean class, Primary key class	
Features of Entity beans	Entity beans survive failures , Entity bean instances are a view into a database	T1: Pg.No. 112- 124	Difference between session and entity beans, Methods ejbLoad(), ejbStore(), Lading and storing process	180
	Several entity bean instances		Threads of execution, Handling	

	may represents the same underlying data		simultaneous client requests, Yhread safe instance, Performance boosting	
	Entity bean instances can be pooled		Decide the fields in database, Callback methods , role of EJB container	
	Two ways to persist entity beans		State save , State load, Persistence API, Container managed persistence	
	Creation and removal of entity beans		Understanding how to create and remove entity beans, Methods in all interfaces	
	Entity beans can be found		Data representation, Relationship between remove() and ejbremove()	
	Modify entity bean data without using EJB		Manual modification, using home object for all manipulation.	
Entity contexts Entity bean coding basics	getEJBLocalObject()/getEJBObject()	T1: Pg.No. 124-135	Code of Context interface, Entitycontext interface	180
	getPrimaryKey()		When to call the function, How to call the function, Methods associated to primarykey(),	
	Finding Existing Entity beans ejbfind()		Entity bean, Enterprise bean interface, Methods like, Setentitycontext(), ejbPostCreate(), ejbLoad()	
The Bean managed persistent Example	Creation of Home, Local interface	T1: Pg.No. 136-156	Take bank example, Create object Model, Write source code of Home, Local interface,	180
	Generate bean		Source code of Account.java with Main methods, such as getBalance(), getAccount(), Business logic methods	
	Create Client program		Get reference to interfaces, Accept the required details, Necessary method invocations	
deployment descriptor, **Running the Clint program**	Setting up the database	T1: Pg.No. 157-166	Create DDL for account table, Find JDBC driver and JNDI locations	180
	Server side output		Obtain output based on EJB container behavior, finder method	
	Client side output		Produced result based on the input in client program and method invocations	
	Bean's life cycle.		Necessary methods for implementation of entity bean , Steps for entire process	
UNIT : IV(LECTURE HOURS:15)				
Features of CMP entity beans	CMP entity beans are Subclassed, No Declared fields	T1: Pg.No. 167-176	Subclassing concept, Examples	180
	Get/Set methods are defined in the subclass, Abstract persistence schema		Abstract Class, get, set methods coding, Deployment descriptor snippet	
	Query language, ejbSelect() methods		Finder methods, The process of developing and deploying a CMP bean	
Implementation Guidelines, CMP example,	List of Methods and its typical implementation	T1: Pg.No. 176-191	Methods such as ejbLoad(), ejbStore(), ejbRemove(), unsetEntitycontext()	180
	local, home, localhome interfaces		Object model, Methods needed for home, Local interfaces	
	Container specific Deployment descriptor		Use custom primary key classes, finally() classes	

Deployment descriptor of CMP, Message driven beans Java Message Service	Running the program, Life cycle of CMP entity bean	T1: Pg.No. 191-210	Run server side program, Give input for client program and obtain the necessary results, Steps involved in life cycle	180
	Motivation to use of Message driven beans		Asynchronous Programming, concern of messaging, RMI vs messaging	
	Messaging domain		Pub/Sub, Point to point (PTP,)	
	JMS API		Client view of JMS sytem, Programming model, JMS interfaces,	
Integrating JMS with EJB, Developing message driven beans	Approaches to integrate JMS with EJB	T1: Pg.No. 211-223	Using java objects, Reuse existing type of EJB component, Characteristics of message driven beans	180
	The semantics		Ejb.messagdrivenbeans(), Messagedrivencontext, necessary methods	
	A simple example		List of methods and their description, Implementation class, Life cycle, Client program	
Advanced concepts, Message driven bean Gotchas	Support Containers	T1: Pg.No. 223-235	Transactions, Load balancing, Duplicate consumption in a cluster	180
	Message Ordering		Message driven beans in a cluster, Queuing concept	
	Missed ejbRemove() calls, Poison Messages		How beans cause poison messages, Strategy to resolve,	
	Return back to message producers		A simple request/response paradigm solution, Advantage and disadvantage of using this architecture,	
UNIT : V(LECTURE HOURS:15)				
Introduction to JDBC	Goals of JDBC	T2: Pg.No. 25-38	Structure of JDBC, Basic class and interfaces, Databases and drivers, Alternatives to JDBC	180
	Connecting to the database		Connection troubles, JDBC classes for connection creation	
	Basic database access		Classes for access, SQL Null vs Java null, Clean up, Modifying the database	
Introduction to JDBC	SQL dtatypes and Java datatypes	T2: Pg.No. 39-56	SQL numeric, SQL long, JDBC prescribed sql datatypes	180
	Scrollable result sets		Result set types, result set navigation, Driver with scrollable result set	
	The JDBC support classes		Java.sql.types, SQLexception, SQLwarning and Datatruncation	
	A database servlet		Getting configuration information, Random visitor comments, Saving, generating and inserting new comments	
Advanced JDBC	Prepared SQL	T2: Pg.No. 57-81	Prepared statements, Stored Procedures,	180
	Batch processing		Kind of statements to use, Efficient approach, Methods for batch processing	

	Updatable result sets		Update, deletes, Inserts, Visibility of changes, Refreshing data from database	
	Advanced data types		Blobs and Clobs,Arrays, SQL3 types, Type mapping	
	Meta-data		Result set, database meta-data,	
The JDBC optional package	Data sources	T2: Pg.No. 92-101	Naming and directory services, JDBC data sources, JNDI service provider	180
	connection pooling		Working Model , Regular connection API, Open, talk, close connection	
	Row sets		Resultset interface, Swing application, Configuration, Usage, Row set events	
	Distributed transactions		Data sources, Sample program	
Other enterprise APIs, Structure of RMI, An object server, EJB roles, Kinds of beans.	Java naming and directory interface	T2: Pg.No. 105-125	Object binding, Object lookup	180
	Remote method invocation		Structure of RMI, Remote interfaces, Object Server, Stubs and skeleton	
	Object serialization		String, Hashmap, Example SerialDemo	
	Enterprise javabeans		EJB roles, Kinds of beans, Delimits of RMI	

Text Books:

1. Mastering Enterprise JavaBeans| Edition :2|Wiley Student Edition| Ed Roman, Scott Ambler, Tyler Jewell
2. Database Programming with JDBC and Java |Edition:2|SPD O`REILLY |George Reese