VIGNAN'S INSTITUTE OF INFORMATION TECHNOLOGY (AUTONOMOUS)

VISAKHAPATNAM

ACADEMIC REGULATIONS for MCA (Regular)

(Applicable for the batches admitted from 2017 onwards)

Academic Regulations – MCA Regular Program

1.0 Eligibility For Admissions

1.1 Admission to the MCA program shall be made subject to the eligibility, qualifications and specialization prescribed by the University from time to time.

1.2 Admissions shall be made on the basis of merit rank obtained by the qualifying candidate at ICET examination or at an Entrance Test conducted by the University subject to reservations prescribed by the university from time to time.

Note: Recognized Bachelor's Degree of minimum 3 years duration with Mathematics at 10+2 level or Graduate Level.

Award of MCA Degree

1.1 A student shall be declared eligible for the award of the MCA degree, if he pursues a course of study and completes it successfully for not less than three academic years and not more than six academic years.

1.2 A student, who fails to fulfill all the academic requirements for the award of the degree within six academic years from the year of his admission, shall forfeit his seat in MCA course.

1.3 The minimum clear instruction days for each semester shall be 90 days.

1.4 A student shall register for courses in each semester as per the courses offered by the department.

2.0 Attendance

2.1 A candidate shall be deemed to have eligibility to appear for end semester examinations, if he has put in a minimum of 75 percent of attendance in aggregate of all the subjects.

2.2 Condonation of shortage of attendance up to 10 percent (i.e. 65 percent and above and below 75 percent) may be given by the committee consisting of Principal, Head of the Department and two senior faculty members.

2.3 Condonation of shortage attendance shall be granted only on genuine and valid reasons on representation by the candidate with supporting evidence.

2.4 Shortage of attendance below 65 percent shall in no case be condoned.

2.5 A student shall not be promoted to the next semester unless he/she fulfills the attendance requirements of the previous semester.

3.0 Evaluation And Academic Requirements

The performance of the candidate in each semester shall be evaluated subject-wise, with a maximum of 100 marks for theory and 100 marks for practicals, on the basis of the Internal Evaluation and End Semester Examination.

3.1 For the theory subjects 60 marks shall be awarded based on the performance in the External End Examination, Remaining 40 marks shall be awarded based on the Internal marks Evaluation. Internal Evaluation shall be made based on average of the marks secured in the two mid Term-Examinations conducted one in the middle of the term examination and the other immediately after the completion of instruction. Each midterm examination shall be conducted for duration of 120 minutes with 4 questions to be answered out 4 questions.

3.2 For practical subjects, 60 marks shall be awarded based on the performance in the End semester examinations, 40 marks shall be awarded based on the day-to-day performance as Internal Marks.

3.3 There shall be a seminar presentation in VI semesters. For technical seminar there will be only internal evaluation of 100 marks. A student has to secure a minimum of 50 marks or else the student has to repeat the technical seminar presentation. The assessment will be made by a Board consisting of Head of the Department and two other senior staff members of the department.

3.4 A candidate shall be deemed to have secured the minimum academic requirements in a subject if he secures a minimum of 40% of marks in the End Examination and a minimum aggregate of 50% of the total marks in the End semester Examination and Internal Evaluation taken together. If a candidate secures a minimum of 40% of marks in the End examination and 40-49% of the total marks in the end semester examination and internal evaluation taken together and secures an overall aggregate of 50% in the total semester he may be passed in those subjects also.

3.5 A candidate shall be given one chance to re-register for each subject provided the internal marks secured by a candidate are less than 50% and he has failed in the end examination. The re-registration is permitted only after completion of his regular course and before submitting his final project report and as and when the subjects (or any other equivalent subject as decided by the committee) are offered. In the event of taking another chance, the internal marks and end examination marks obtained in the previous attempt are nullified. At a given time a candidate is permitted to re-register for a maximum of 2 subjects.

3.6 A student shall be allowed to submit the project report only after fulfilling the attendance requirements of all the semesters. The viva-voce examination shall be conducted at the end of the course work and after the completion of the end semester examination of the final semester.

3.7 Laboratory examination for MCA course must be conducted with two examiners, one of them being Laboratory Class teacher and second examiner shall be other than Class Teacher.

3.8 The Technical Seminar shall be conducted in the third year sixth semester. For Technical Seminar, a student under the supervision of faculty member, shall collect the literature on a topic and critically review the literature and submit it to the department in the form of a report and shall make an oral presentation before the Departmental Committee. The Departmental Committee consists of Head of the Department, Seminar Supervisor and one senior faculty member of the department. For Technical seminar, there will be only internal evaluation for 100 marks and a student has to secure minimum of 50 marks to be declared successful.

4.0 Evaluation Of Project Work

4.1 Mini Project:

There shall be a Mini project presentation in IV semester. For presentation there will be only internal evaluation of 100 marks. A student has to secure a minimum of 50% or else the student has to repeat the presentation. The assessment will be made by a Board consisting of Head of the Department and two other senior staff members of the department.

4.2 Major Project:

5.2.1 A Project review committee (PRC) of the college is to be constituted with principle as chairperson, Head of the department, which are offering PG course and two other senior faculty members of the departments offering MCA.

5.2.2 Registration of Project work: A student is permitted to register for the project work after satisfying the attendance requirement of all the courses (theory and practical courses) up to V semester.

5.2.3 Every student shall work on project approved by the PRC of the college. (or) A student can undergo for internship and produce the project report to the PRC for external evaluation.

5.2.4 The duration of the project is for one semester.

5.2.5 Three copies of thesis, certified by the supervisor shall be submitted to the college/school/institute.

5.2.6 The project report shall be adjudicated by the one external examiner selected by the Controller of Examinations out of 5-member panel submitted by the department who are eminent in that field of study.

5.2.7 The viva-voce examination of the project report shall be conducted by a board consisting of the external examiner (See section 4.7), the head of Department and Supervisor. The board shall jointly report student work as :

A. Excellent
B. Good
C. Satisfactory
D. Unsatisfactory

Head of the department shall co-ordinate and make arrangements for conduct of viva-voce examination.

5.2.8 If the report of the viva-voce is not satisfactory, the student will re-take the viva-voce examination after 3 months. If he fails to get a satisfactory report at the second viva-voce examination, he will not be eligible for the award of the degree, unless the student is asked to revise and re-submit. If the report of the examiner is unfavorable again, the project shall be summarily rejected.

6.0 Award of Degree And Class

After a student has satisfied the requirements prescribed for the completion of the program, the student is eligible for the award of MCA degree. The student shall be considered in one of the following four classes.

Class Awarded	% of marks to be secured
First Class with Distinction	70% and above
First Class	Below 70% but not less than 60%
Second Class	Below 60% but not less than 50%
Third Class	Below 50% but not less than 40%

(The marks in Internal Evaluation and the examination shall be shown separately in the marks memorandum)

7.0 With Holding Of Results

If the student has not paid any dues to the Institute or any case of indiscipline is pending against him, the result of the student will be withheld and he will not be allowed in the next semester. The issue of degree is liable to be withheld in such cases.

8.0 Transitory Regulations

8.1 A student has to continue the course work along with the regular students of the respective semester in which the student gets readmission.

8.2 A student has to register for substitute / compulsory subjects offered in place of subjects studied earlier.

8.3 The mode of internal evaluation and end-semester examinations shall be on par with the regular
Students, i.e. the student has to follow the mode of internal evaluation and then question paper
model for the end-semester examinations along with the regular students of the respective semester in which the student gets readmission. The marks secured in the internal and end-semester examinations will be in accordance with the regulations under which the student was first admitted.

8.4 For the courses studied under earlier regulations but failed, the student has to appear, pass from
the supplementary examinations as and when conducted. The question paper model shall remain same as the one in which the student took examination during previous regulations.

8.5 The promotion criteria based on attendance with the regulations under which the student was first admitted.

8.6 All other academic requirements shall be in accordance with the regulations under which the student was first admitted.

8.7 When student is detained due to lack of shortage of attendance, he may be re-admitted when the semester is offered after fulfillment of academic regulations. Whereas the academic regulations hold good with the regulations he/she first admitted.

8.8 The decision of the principal is final on any other clarification in this regard.

9.0 General

9.1 The academic regulations should be read as a whole for purpose of any interpretation.

9.2 In case of any doubt or ambiguity in the interpretation of the above rules, the decision of the Academic Council is final.

9.3 The Academic Council may charge or amend the academic regulations and syllabus at any time and the changes and amendments made shall be applicable to all the students with effect from the date notified by the Council.

9.4 Wherever the word he, him or his occurs, it will also include she, her and hers.

10. Disciplinary Action for Malpractices

S. No	Nature of Malpractices/ Improper conduct	Punishment
1 (a)	If the candidate possesses or keeps accessible in examination hall, any paper, note book, programmable calculators, Cell phones, pager, palm computers or any other form of material concerned with or related to the subject of the examination (theory or practical) in which he is appearing but has not made use of (material shall include any marks on the body of the candidate which can be used as an aid in the subject of the examination)	Expulsion from the examination hall and cancellation of the performance in that subject only.
(b)	If the candidate gives assistance or guidance or receives it from any other candidate orally or by any other body language methods or communicates through cell phones with any candidate or persons in or outside the exam hall in respect of any matter.	Expulsion from the examination hall and cancellation of the performance in that subject only of all the candidates involved. In case of an outsider, he will be handed over to the police and a case is registered against him.

2	If the candidate has copied in the examination hall from any paper, book, programmable calculators, palm computers or any other form of material relevant to the subject of the examination (theory or practical) in which the candidate is appearing.	Expulsion from the examination hall and cancellation of the performance in that subject and all other subjects the candidate has already appeared including practical examinations and project work and shall not be permitted to appear for the remaining examinations of the subjects of that Semester/year. The Hall Ticket of the candidate is to be cancelled and sent to the University.
3	If the candidate impersonates any other candidate in connection with the examination.	The candidate who has impersonated shall be expelled from examination hall. The candidate is also debarred and forfeits the seat. The performance of the original candidate, who has been impersonated, shall be cancelled in all the subjects of the examination (including practicals and project work) already appeared and shall not be allowed to appear for examinations of the remaining subjects of that semester/year. The candidate is also debarred for two consecutive semesters from class work and all University examinations. The continuation of the course by the candidate is subject to the academic regulations in connection with forfeiture of seat. If the imposter is an outsider, he will be handed over to the police and a case is registered against him.
4	If the candidate smuggles in the Answer book or additional sheet or takes out or arranges to send out the question paper during the examination or answer book or additional sheet, during or after the examination.	Expulsion from the examination hall and cancellation of performance in that subject and all the other subjects the candidate has already appeared including practical examinations and project work and shall not be permitted for the remaining examinations of the subjects of that semester/year. The candidate is also debarred for two consecutive semesters from class work and all University examinations. The

		continuation of the course by the candidate is subject to the academic regulations in connection with forfeiture of seat.
5	If the candidate uses objectionable, abusive or offensive language in the answer paper or in letters to the examiners or writes to the examiner requesting him to award pass marks.	Cancellation of the performance in that subject.
6	If the candidate refuses to obey the orders of the Chief Superintendent/Assistant - Superintendent / any officer on duty or misbehaves or creates disturbance of any kind in and around the examination hall or organizes a walk out or instigates others to walk out, or threatens the officer-in charge or any person on duty in or outside the examination hall of any injury to his person or to any of his relations whether by words, either spoken or written or by signs or by visible representation, assaults the officer-in-charge, or any person on duty in or outside the examination hall or any of his relations, or indulges in any other act of misconduct or mischief which result in damage to or destruction of property in the examination hall or any part of the College campus or engages in any other act which in the opinion of the officer on duty amounts to use of unfair means or misconduct or has the tendency to disrupt the orderly conduct of the examination.	In case of students of the college, they shall be expelled from examination halls and cancellation of their performance in that subject and all other subjects the candidate(s) has (have) already appeared and shall not be permitted to appear for the remaining examinations of the subjects of that semester/year. The candidates also are debarred and forfeit their seats. In case of outsiders, they will be handed over to the police and a police case is registered against them.
7	If the candidate leaves the exam hall taking away answer script or intentionally tears of the script or any part thereof inside or outside the examination hall.	Expulsion from the examination hall and cancellation of performance in that subject and all the other subjects the candidate has already appeared including practical examinations and project work and shall not be permitted for the remaining examinations of the subjects of that semester/year. The candidate is also debarred for two consecutive semesters from class work and all University examinations. The continuation of the course by the candidate is subject to the academic regulations in connection with forfeiture of seat.

8	If the candidate possesses any lethal weapon or firearm in the examination hall.	Expulsion from the examination hall and cancellation of the performance in that subject and all other subjects the candidate has already appeared including practical examinations and project work and shall not be permitted for the remaining examinations of the subjects of that semester/year. The candidate is also debarred and forfeits the seat.
9	If student of the college, who is not a candidate for the particular examination or any person not connected with the college indulges in any malpractice or improper conduct mentioned in clause 6 to 8.	Student of the college, expulsion from the examination hall and cancellation of the performance in that subject and all other subjects the candidate has already appeared including practical examinations and project work and shall not be permitted for the remaining examinations of the subjects of that semester/year. The candidate is also debarred and forfeits the seat. Person(s) who do not belong to the College will be handed over to police and. a police case will be registered against them.
10	If the candidate comes in a drunken condition to the examination hall.	Expulsion from the examination hall and cancellation of the performance in that subject and all other subjects the candidate has already appeared including practical examinations and project work and shall not be permitted for the remaining examinations of the subjects of that semester/year.
11	Copying detected on the basis of internal evidence, such as, during valuation or during special scrutiny.	Cancellation of the performance in that subject and all other subjects the candidate has appeared including practical examinations and project work of that semester/year examinations.
12	If any malpractice is detected which is not covered in the above clauses 1 to 11 shall be reported to the University for further action to award suitable punishment.	---

10.1 Malpractices identified by squad or special invigilators

10.1. Punishments to the candidates as per the above guidelines.

11. UGC recommended punishment for ragging

11.1 Suspension from attending classes and academic privileges.

11.2 Withholding/withdrawing scholarships/fellowship and other benefits.

11.3 Debarring from appearing in any test/examination or other evaluation process

11.4Withholding results.

11.5 Debarring from representing the institution in any regional, national or international meet, tournament, youth festival etc.

11.6 Suspension/expulsion from the hostel.

11.7 Cancellation of admission.

11.8 Rustication from the institution for period ranging from 1 to 4 semesters.

11.9 Expulsion from the institution and consequent debarring from admission to any other Institution for a specified period.

11.10 Fine may extend up to Rs. 2.5 lakhs.

12.0 MCA PROGRAM STRUCTURE

MCA 1st Year (I-Semester)

Course Code	Theory/Lab	T	P	C	CE	SE	TO
4010171101	C Programming and Data Structures	4+1	0	4	40	60	100
4010171102	Computer Organization	4+1	0	4	40	60	100
4010171103	Discrete Mathematical Structures and Graph Theory	4+1	0	4	40	60	100
4010171104	Statistical Programming with R	4+1	0	4	40	60	100
4010171105	Accounting and Financial Management	4+1	0	4	40	60	100
4010171121	English Language Communication Skills Lab	0	3	2	40	60	100
4010171122	C Programming and Data Structures Lab	0	3	2	40	60	100
4010171123	Statistical Programming with R Lab	0	3	2	40	60	100
	Total	**25**	**9**	**26**	**320**	**480**	**800**

MCA 1st Year (II-Semester)

Course Code	Theory/Lab	T	P	C	CE	SE	TO
4010171201	OOPS Through JAVA	4+1	0	4	40	60	100
4010171202	Operating Systems	4+1	0	4	40	60	100
4010171203	Software Engineering	4+1	0	4	40	60	100
4010171204	Operations Research for Computer Applications	4+1	0	4	40	60	100
4010171205	Computer Graphics	4+1	0	4	40	60	100
4010171221	OOPS Through JAVA Lab	0	3	2	40	60	100
4010171222	Web Technologies Lab	0	3	2	40	60	100
4010171223	Operating System & Computer Graphics Lab	0	3	2	40	60	100
	Total	**25**	**9**	**26**	**320**	**480**	**800**

MCA 2nd Year (III-Semester)

Course Code	Theory/Lab	T	P	C	CE	SE	TO
4010172301	Database Management Systems	4+1	0	4	40	60	100
4010172302	Advanced JAVA Programming	4+1	0	4	40	60	100
4010172303	UNIX Programming	4+1	0	4	40	60	100
4010172304	Management Information System	4+1	0	4	40	60	100
4010172305	Design and Analysis of Algorithms	4+1	0	4	40	60	100
4010172321	Database Management Systems Lab	0	3	2	40	60	100
4010172322	UNIX Programming Lab	0	3	2	40	60	100
4010172323	Advanced JAVA Programming Lab	0	3	2	40	60	100
	Total	**25**	**9**	**26**	**280**	**480**	**800**

MCA 2nd Year (IV-Semester)

Course Code	Theory/Lab	T	P	C	CE	SE	TO
4010172401	Object Oriented Analysis and Design	4+1	0	4	40	60	100
4010172402	Computer Networks	4+1	0	4	40	60	100
4010172403	Data warehousing and Mining	4+1	0	4	40	60	100
4010172404 – 4010172406	**Elective-1**	4+1	0	4	40	60	100
4010172407 - 4010172409	**Elective-2**	4+1	0	4	40	60	100
4010172421	Soft Skills Lab	0	3	2	40	60	100
4010172422	Data Warehousing and Mining Lab	0	3	2	40	60	100
4010172423	Object Oriented Analysis and Design Lab	0	3	2	40	60	100
4010172431	Mini Project	0	0	2	0	0	100
	Total	**25**	**9**	**28**	**320**	**480**	**900**

Elective -1

4010172404	:	Mobile Computing
4010172405	:	Human Computer Interaction
4010172406	:	Cloud Computing

Elective -2

4010172407	:	Software Project Management
4010172408	:	Artificial Intelligence
4010172409	:	Embedded Systems

MCA 3rd Year (V-Semester)

Course Code	Theory/Lab	T	P	C	CE	SE	TO
4010173501	Big Data Analytics	4+1	-	4	40	60	100
4010173502	Network Programming	4+1	-	4	40	60	100
4010173503	Python Programming	4+1	-	4	40	60	100
4010173504 - 4010173506	**Elective-3**	4+1	-	4	40	60	100
4010173507 – 4010173509	**Elective-4**	4+1	-	4	40	60	100
4010173521	Big Data Analytics Lab	-	3	2	40	60	100
4010173522	Network Programming Lab	-	3	2	40	60	100
4010173523	Python Programming Lab	-	3	2	40	60	100
	Total	**25**	**9**	**26**	**320**	**480**	**800**

Elective -3

4010173504	:	Cyber Security
4010173505	:	Computer Forensics
4010173506	:	E-Commerce

Elective -4

4010173507	:	Internet of Things
4010173508	:	Multimedia Application Development
4010173509	:	Software Testing Methodologies

MCA 3rd Year (VI-Semester)

Course Code	Theory/Lab	T	P	C	CE	SE	TO
4010173632	Technical Seminar	-	-	2	-	-	100
4010173637/ 4010173638	Internship /Major Project	-	-	10	-	-	-
	Dissertation/Thesis Excellent/Good/Satisfactory/Not Satisfactory						

T: Theory P: Practical C: Credits

CE: Continuous Evaluation

SE: Semester End Evaluation

To: Total Marks

I MCA DETAILED SYLLABUS
(1st & 2nd Semesters)

I MCA I SEM Subject Code: MC1711 Subject Name: C Programming and Data Structures

Course Objectives:

1) Students will learn how to analyze the problem and writing a code for given algorithm

2) Students will learn different operators, control structures and familiarize with the large number of built-in functions available in C language

3) Students will learn the various data structure concepts like searching, sorting, stack, queue etc.

Course Prerequisites:

Students should have knowledge of

1) Basic Computer and programming knowledge.
2) Basic knowledge in Mathematics
3) Reasoning and Logical thinking

Course Outcome:

Students will be able to:

1) Understand the basic C programming
2) Knowledge on arrays, strings and functions
3) Knowledge on pointers, structures and unions
4) Summarize recursive methods and compare iterative and recursive solutions for elementary problems
5) Illustrate the usage of various data structures and familiarize with various searching and sorting techniques
6) Solve problems using various data structures like linear list, stack, queue, tress and graphs

UNIT-I **C BASICS AND CONTROL STRUCTURES**

COMPUTER BASICS: Introduction to Computers, HW and SW Concepts, Problem Solving Techniques, Program Development Steps, **C BASICS**: Basic structure of C Program, History of C, identifiers, basic data types and sizes, constants, variables, different types of operators, type conversions, expressions, precedence and order of evaluation. **CONTROL STRUCTURES:** if statement- switch and go to statement, labels and looping statements.

UNIT-II **ARRAYS, STRINGS AND FUNCTIONS**

ARRAYS: Basic concepts of arrays- declaration, definition, accessing elements, storing elements, 1-D arrays, 2-D arrays, **STRINGS:** String concepts, String manipulations, character arrays, multi-dimensional arrays, and string handling functions. **FUNCTIONS:** Basic concepts on functions, types of functions, categories of functions, Function parameters, Parameter passing techniques, scope rules, block structure, Storage classes, Recursion concept, recursive functions.

UNIT-III **POINTERS, STRUCTURES AND UNIONS**

POINTERS: Definition, declaration and initialization of pointer variables, the & and * operators, void pointer, pointer-to-pointer, pointers and function arguments, passing by address, character pointer and functions, pointer and multidimensional arrays, dynamic memory management functions, array of pointer, command line arguments.

STRUCTURES and UNIONS: Definition, initialization, accessing structure elements, array of structures and array within the structure, nested structures, self-referential structure, Union's concept, differences between structures and unions, type-def, bit fields, enum data type.

FILES: Concepts of a files, text and binary files, opening and closing a file, file I/O operations, formatted I/O operations.

UNIT-IV **DATA STRUCTURES : STACKS AND QUEUES**
Hours

DATA STRUCTURES: Introduction to Data Structures, Time complexity, Space complexity, Pattern matching, naïve method, Robin Karp Algorithm-Searching-Linear and Binary Search methods, Sorting- Bubble sort, selection sort, insertion sort, quick sort, merge sort.

STACKS and QUEUES: Stack concept, applications of stack-infix to postfix conversion, postfix expression evaluation, implementation of stack using array- Queue concept, types of queues, applications of queues, queue implementation using array**.**

UNIT-V **LINKED LISTS, TREES AND GRAPHS**

LINKED LISTS: Singled linked lists, doubly linked lists, circular linked list, linked stack and linked queue. **TREES:** Basic concepts of trees, binary trees, terminology, representation, traversals, **GRAPHS:** Graphs- terminology, representation, graph traversals, (dfs & bfs)-Dijkstra-Kruskal-Prims Algorithms only.

Text Books:

1) C and Data Structures: A snapshot oriented treatise using live engineering examples, N B Venkateswarlu, E. V Prasad, S Chand & Co.

2) Computer science, A structured programming approach using C, B.A. Forouzan and R. F. Gilberg, Third edition, Thomson.

Reference Books:

1) Fundamentals of Data Structures in C , Horowitz, Sahni, Anderson- Freed, 2nd ed., universities Press, 2008.

2) The C Programming Language, B.W. Kernighan, Dennis M. Ritchie, PHI/ Pearson.

3) Data Structures Using C , A. S. Tanenbaum, Y. Langsam, and M.J. Augenstein, PHI/ Pearson

4) C Programming and Data Structures, E Balaguruswamy, TMH, 2008.

I MCA I SEM Subject Code: MC1712 Subject Name: Computer Organization

Course Objectives:

1) Students will learn the basics of designing and basic structure of computer Organization.

2) Students will learn different type Machine instruction and assembly program

3) Students will learn the key concept of hard ware and software related design issue with Computer Organization.

Course Prerequisites:

Students should have knowledge of

1) Basic function and structure of CO and its design process.

2) Basic Machine languages and type of instructions.

3) Input-Output instruction and memory system.

Course Outcome:

Students will be able to:

1. The basic components of a computer, including CPU, memories, and input/output, and their organization representation of data, addressing modes, instructions sets.
2. Discriminate different register transfer micro operations, Principles of hardwired and micro programmed control
3. Demonstrate various fixed and floating point arithmetic operations, performing the Arithmetic operations of binary number systems and formulating the arithmetic functions and solve scientific problems by means of a numerical analysis method
4. Extrapolate memory organization and input, output organizations
5. Generalize pipe line and vector processing, multi processors and its applications.

UNIT-I Basic Structure Of Computers

Functional unit, Basic Operational concepts, Bus structures, System Software, Performance, The history of computer development.

UNIT-II Machine Instruction and Programs

Instruction and Instruction Sequencing: Register Transfer Notation, Assembly Language Notation, Basic Instruction Types, Addressing Modes, Basic Input/output Operations, The role of Stacks and Queues in computer programming equation. Component of Instructions: Logic Instructions, shift and Rotate Instructions.

UNIT-III Type of Instructions

Arithmetic and Logic Instructions, Branch Instructions, Addressing Modes, Input/output Operations.

UNIT-IV Input / Output Organization

Accessing I/O Devices, Interrupts: Interrupt Hardware, Enabling and Disabling Interrupts, Handling Multiple Devices, Direct Memory Access, Buses: Synchronous Bus, Asynchronous Bus, Interface Circuits, Standard I/O Interface: Peripheral Component Interconnect (PCI) Bus, Universal Serial Bus (USB).

UNIT-V The Memory Systems

Basic memory circuits, Memory System Consideration, Read- Only Memory: ROM, PROM, EPROM, EEPROM, Flash Memory, Cache Memories: Mapping Functions, INTERLEAVING **Secondary Storage:** Magnetic Hard Disks, Optical Disks, **Processing Unit:** Fundamental Concepts: Register Transfers, Performing An Arithmetic Or Logic

Operation, Fetching A Word From Memory, Execution of Complete Instruction, Hardwired Control,

Micro programmed Control: Microinstructions, Micro program Sequencing, Wide Branch Addressing Microinstructions with next –Address Field.

Text Books:

1) Computer Organization, Carl Hamacher, Zvonks Vranesic, Safea Zaky, 5th Edition, McGraw Hill.
2) Computer Architecture and Organization , John P. Hayes ,3rd Edition, McGraw Hill.

Reference Books:

1) Computer Organization and Architecture – William Stallings Sixth Edition, Pearson/PHI.
2) Structured Computer Organization – Andrew S. Tanenbaum, 4th Edition PHI/Pearson.
3) Fundamentals or Computer Organization and Design, - Sivaraama Dandamudi Springer Int. Edition.

I MCA I SEM: DISCRETE MATHEMATICAL STRUCTURES AND GRAPH THEORY – MC1713

UNIT- I: Mathematical Logic: Statements and notations, Connectives, Well formed formulas, Truth tables, tautology, equivalence implication, Normal forms, Theory of inference for the statement calculus, Rules of inference, Consistency of premises and indirect method of proof, Predicate calculus: Predicates, statement functions, variables and quantifiers, predicate formulas, free & bound variables, universe of discourse, inference theory of predicate calculus

UNIT-II: Relations: Introduction, Relations and ordering, Properties of binary Relations, Equivalence, Compatibility Relations, Partial ordering, Hasse diagram, Lattice and its Properties. Functions: composition of functions, Inverse Function, Recursive Functions.

UNIT-III: Elementary Combinatorics: Basis of counting, Enumeration of Combinations & Permutations, Enumerating of Combinations & Permutations with repetitions and constrained repetitions, Binomial Coefficients, Binomial Multinomial theorems, principles of Inclusion – Exclusion.

UNIT-IV:

Recurrence Relations: Generating Function of Sequences, Calculating Coefficient of generating functions, Recurrence relations, Solving recurrence relation by substitution and Generating functions, The method of Characteristic roots, Solution of Inhomogeneous Recurrence Relation.

UNIT-V:

Graph Theory: Representation of Graph, Spanning Trees, BFS, DFS, Kruskals Algorithm, Binary trees, Planar Graphs, Graph Theory and Applications, Basic Concepts, Isomorphism and Sub graphs, Multi graphs and Euler circuits, Hamiltonian graphs.

TEXTBOOKS:

i) Discrete Mathematical Structures with Applications to computer science J.P Tremblery, R. Manohar, TMH
ii) Discrete Mathematical for computer Scientists & Mathematicians " J.L. Molt, A.Kandel, T. P. Baker, PHI

REFERENCE TEXTBOOKS:

1. Elements of Discrete Mathematics, C L Liu, D P Mohanpatra,TMH
2. Discrete Mathematics, Schaum's Outlines, Lipschutz, Lipson, TMH.
3. Discrete Mathematical Structures, Kolman, Busby, Ross, 6th ed., PHI, 2009
4. Discrete Mathematics, Johnsonbaugh, 6th ed., Pearson, 2005
5. Discrete Mathematics, Malik, Sen, 6th ed., Cengage Learning, 2004
6. Discrete Mathematics for computer science, Bogart, Stein and Drysdale, Springer, 2005

I MCA I SEM	**Subject Code: MC1714**	**Subject Name:** **STATISTICAL PROGRAMMING WITH R**

Course Objectives:

1) Students will learn how to write programs in R Language

2) Students will learn control structures, graphics functions in R

3) Students will learn how to Implement Statistical Concepts in R

Course Prerequisites:

Students should have knowledge of

1) Basic knowledge on R Programming
2) Basic knowledge in Probability concepts
3) Basic Programming knowledge.

Course Outcome:

Students will be able to:

1) understand the basic s of R Programming
2) knowledge on R Programming Control statements and functions
3) knowledge on Graphics
4) Awareness on Statistical Concepts

UNIT-I Introduction to R

Introduction, How to run R, R Sessions and Functions, Basic Math, Variables, Data Types, Vectors, Conclusion, Advanced Data Structures, Data Frames, Lists, Matrices, Arrays, Classes**.**

UNIT-II R Programming Control Statements

R Programming Structures, Control Statements, Loops, - Looping Over Non vector Sets,-If-Else, Arithmetic and Boolean Operators and values, Default Values for Argument, Return Values, Deciding Whether to explicitly call return- Returning Complex Objects, Functions are Objective, No Pointers in R, Recursion, A Quick sort Implementation-Extended Extended Example: A Binary Search Tree.

UNIT-III R Programming Functions

Doing Math and Simulation in R, Math Function, Extended Example Calculating Probability-Cumulative Sums and Products-Minima and Maxima- Calculus, Functions Fir Statistical Distribution, Sorting, Linear Algebra Operation on Vectors and Matrices, Extended Example: Vector cross Product- Extended Example: Finding Stationary Distribution of Markov Chains, Set Operation, Input /output, Accessing the Keyboard and Monitor, Reading and writer Files.

UNIT-IV Graphics

Graphics, Creating Graphs, The Workhorse of R Base Graphics, the plot () Function – Customizing Graphs, Saving Graphs to Files.

UNIT-V Probability Concepts

Probability Distributions, Normal Distribution- Binomial Distribution- Poisson Distributions

Other Distribution, Basic Statistics, Correlation and Covariance, T-Tests,-ANOVA. Linear Models, Simple Linear Regression, -Multiple Regression Generalized Linear Models, Logistic Regression, - Poisson Regression- other Generalized Linear Models-Survival Analysis, Nonlinear Models, Spines- Decision- Random Forests.

Text Books:

1. The Art of R Programming, Norman Matloff, Cengage Learning
2. R for Everyone, Lander, Pearson

Reference Books:

1) R Cookbook, PaulTeetor, Oreilly.

2) R in Action, Rob Kabacoff, Manning

I MCA I SEM: ACCOUNTING AND FINANCIAL MANAGEMENT – MC1715

Course Description:

Management accounting or managerial accounting is the process of identifying, analyzing, recording and presenting financial information that is used for internally by the management for planning, decision making and control. In contrast to financial accounting, managerial accounting is concerned with providing helpful information and reports to internal users such as managers and entrepreneurs etc. so that they can control and plan the business activities.

Financial management refers to the efficient and effective management of money (funds) in such a manner as to accomplish the objectives of the organization. It is the specialized function directly associated with the top management. The significance of this function is not seen in the 'Line' but also in the capacity of 'Staff' in overall of a company. It has been defined differently by different experts in the field. The term typically applies to an organization or company's financial strategy, while personal finance or financial life management refers to an individual's management strategy.

Course Objectives & Outcomes:

Course Objectives:

The objectives of the course are:

1. To learn the concept and the role of accounting.
2. To know the systems of accounting.
3. To understand and the Preparation of trail balance – Final accounts.
4. To understand the concept of financial management.
5. To learn the concept of cost of capital.
6. To understand the capital structure
7. To able to know the cost accounting.
8. To learn the total computerized accounting system.

Course Outcomes:

After completing this Course, the student should be able to:

1. To identify the need and the role of accounting in present modern business.
2. To have capabilities to preparation of trail balance – Final accounts.

3. Financial management role and objectives of the business.
4. To explain the Importance of the cost behavior.
5. Use of the standard costing and budgeting in present business level.
6. Introduction and importance of computerized accounting system in present business scenario

UNIT-I:

Introduction of accounting: Meaning of accounting-Book keeping Vs Accounting - – Objectives of accounting- Users of accounting information-Branches of accounting.

Definition of financial accounting – Principles of accounting (GAAP) - Classification of accounts-Accounting Cycle – Preparation of Book of Original entry, Ledger, Trial balance – Final
Accounts with Simple adjustments.

UNIT-II:

Financial Management: Nature and Scope and Objectives of Financial Management. Firms Missions and Objectives-function of FM- Profit Maximization Vs Wealth maximization Vs EPS Maximization – Role of Financial Manager.

UNIT–III:

Cost accounting: Meaning of cost - Elements of cost - Methods of Costing, Techniques of Costing - Role of Cost accounting - Financial accounting Vs Cost accounting Vs Management Accounting – Concept of BEP analysis – CVP Analysis.

UNIT-IV:

Standard costing and budgeting : Meaning - Nature, scope of Standard Costing and the concept of variance analysis – materials variance, labor variance and sales variance – The concept of Budget - cash budget, sales - budget – flexible Budgets, master budgets.

UNIT-V:

Introduction to Computerized Accounting System: Definition - Importance - Merits and Demerits of CAS – Importance coding logic and codes, master files, transaction files-Documents used for data collection, processing of different files and Outputs obtained.

TEXT BOOKS:

1 Accounting for Managers, P. Vijaya Kumar, and Himalaya Publications.
2. Accounting for Management. Vijaya Kumar.TMH.
3. Financial Accounting, S.N Maheswari and S.K. Maheswari, Vikas.
4. Financial Accounting, A. Mukherjee and M. Heneef, TMH.

I MCA I SEM: ENGLISH LANGUAGE COMMUNICATION SKILLS LAB – MC1716

Course objectives

- understand the fundamental principles of effective communication;
- apply the critical and creative thinking abilities necessary for effective communication in today's modern world;
- organize and express ideas in writing and speaking to produce messages suitably tailored for the topic, objective, audience, communication medium and context; and
- Demonstrate clarity, precision, conciseness and coherence in your use of language.

Course Outcomes:

- Use English language fluently, accurately and appropriately.
- Discuss and discover barriers to effective communication.
- Demonstrate skills in listening comprehension, GDs and Interview.
- Read and answer questions (orally and in writing) based on passages.
- Show effective writing skills in academic and professional contexts.

Syllabus:

1. Group Discussion – dynamics of group discussion, Lateral thinking, Brainstorming, Mind Mapping.

2. **Cover letter and Resume writing** – structure and presentation, planning, defining the career objective.

3. **Vocabulary building**, Creativity in using Advertisements, Case Studies etc.

4. **Listening comprehension** – Achieving ability to comprehend material delivered at relatively fast speed; comprehending spoken material in Standard Indian English, British English, and American English

5. **Personality Development:** Decision-Making, Problem Solving, Goal Setting, Time Management & Positive Thinking. Work-Life-Balance (Avoidable Stress, Unavoidable stress, Positive Stress)

6. **Cross-Cultural Communication** / Non-Verbal Communication, Problems of Language, Lack of Language equivalency/difficulties in using English.

7. **Writing Skills – Technical Report Writing/ Project Proposals** – Types of formats and styles, subject matter – organization, clarity, coherence and style, planning, data-collection, tools, analysis.- Progress and Project Reports. Essays for competitive examinations, analyzing newspaper articles

8. **Interview Skills** – concept and process, pre-interview planning, opening strategies, answering strategies, interview through Tele and video-conferencing

Recommended Books:

1. Simon Sweeny, English for Business Communication, CUP, First South Asian Edition, 2010.

2. M. Ashraf Rizvi, Effective Technical Communication, Tata McGraw-Hill Publishing Company Ltd. 2005

3. Dr A Ramakrishna Rao, Dr. G. Natanam & Prof SA Sankaranarayanan, English Language Communication:

4. A Reader cum Lab Manual, Anuradha Publications, Chennai, 2006

5. Dr. ShaliniVerma, Body Language- Your Success Mantra S. Chand, 2006.

I MCA I SEM

C PROGRAMMING AND DATA STRUCTURES LAB – MC1717

Course Objectives:

• To learn/strengthen a programming language like C, To learn problem solving techniques

• To introduce the student to simple linear and non linear data structures such as lists, stacks, queues, etc.,

Recommended Systems/Software Requirements:

• Intel based desktop PC, ANSI C Compiler with Supporting Editors, IDE's such as Turbo C, Bloodshed C

Course Outcomes:

- Able to write programs in C Language
- Develop logical and analytical thinking in C
- Knowledge in writing programs in various concepts like arrays, functions, pointer etc.
- How to read and write contents from or into a file

Exercise l

a) Write a C Program to find the sum of individual digits of a positive integer.

b) A Fibonacci sequence is defined as follow: the first second terms in the sequence are 0 and 1. Subsequent terms are found by adding the preceding two terms in the sequence. Write a C Program to generate the first n terms of the sequence.

C) Write a C Program to generate all the prime numbers between 1 and n, where n is a value supplied by the user.

d) Write a C Program which checks a given integer is Fibonacci number or not.

Exercise 2

a) Write a C Program to calculate the following sum:

$sum=a-x^2/2!+x^4/4!-x^6/6!+x^8/8!-x^{10}/10!$

b) Write a C program to find the roots of a quadratic equation.

c) Write a C Program to implement Newton Raphson method for a quadratic equation

Exercise 3

a) Write a C Program that use both recursive and non-recursive functions

(i) To find the factorial of a given number

(ii) To find the GCD (greatest common divisor) of two given integers.

(iii)To Solve Towers of Hanoi problem

(iv) To generate Fibonacci sequence

Exercise 4

a) The total distance travelled by vehicle in 't' seconds is given by distance = $ut+1/2at^2$

where ‘u’ and ‘a’ are the initial velocity (m/sec.) and acceleration (m/sec2). Write a C Program to find the distance travelled at regular intervals of time given the values of ‘u’ and ‘a’. The program should provide the flexibility to the user to select his own time intervals and repeat the calculations values of ‘u’ and ‘a’.

b) Write a C program, which takes two integer operands and one operator form the user, performs the operation and then prints the result. (Consider the operators +,-,*, /, % and use Switch Statement)

Exercise 5

a) Write a C program to find both the largest and smallest number in a list of integers.

b) Write a C program that uses functions to perform the following:

i) Addition of Two Matrices.

ii) Multiplication to Two Matrices.

iii) Checking symmetricity of a square matrix

iv) Calculating transpose of a matrix in-plane manner.

Exercise 6

a) Write a C program to determine if the given string is a palindrome or not.

b) Write a C program to count the lines, words and characters in a given text.

c) Write a C program to swap two numbers using pointers

d) Write a C program to find the reverse of a given string using pointers.

Exercise 7

Write a C program that uses functions to perform the following operations using Structure:

i) Reading a complex number ii) Writing a complex number

iii) Addition of two complex numbers iv) Multiplication of two complex numbers

Exercise 8

a) Write a C program which copies one file to another.

b) Write a C program to reverse the first n characters in a file using command line arguments

Exercise 9

a) Write C program that use both recursive and non recursive functions to perform Linear search for a Key value in a given list.

b) Write C program that use both recursive and non recursive functions to perform Binary search for a Key value in a given list.

c) Write C program that use both recursive and non recursive functions to perform Fibonacci search for a Key value in a given list.

Exercise 10

a) Write C program that implement Bubble sort, to sort a given list of integers in ascending order

b) Write C program that implement Quick sort, to sort a given list of integers in ascending order

c) Write C program that implement Insertion sort, to sort a given list of integers in ascending order

Exercise 11

a) Write C program that implement stack (its operations) using arrays

b) Write C program that implement stack (its operations) using Linked list

Exercise 12

a) Write a C program that uses Stack operations to convert infix expression into postfix expression

a) Write C program that implement Queue (its operations) using arrays.

b) Write C program that implement Queue (its operations) using linked lists

Exercise 13

a) Write a C program that uses functions to create a singly linked list

b) Write a C program that uses functions to perform insertion operation on a singly linked list

c) Write a C program that uses functions to perform deletion operation on a singly linked list

Exercise 14

a) Adding two large integers which are represented in linked list fashion.

b) Write a C program to reverse elements of a single linked list.

c) Write a C program to store a polynomial expression in memory using linked list

d) Write a C program to representation the given Sparse matrix using arrays.

e) Write a C program to representation the given Sparse matrix using linked list

Exercise 15

a) Write a C program to create a Binary Tree of integers

b) Write a recursive C program for traversing a binary tree in preorder, inorder and postorder.

c) Write a non recursive C program for traversing a binary tree in preorder, inorder and postorder.

Exercise 16

a) Write a C program to create a BST

b) Write a C program to insert a node into a BST.

c) Write a C program to delete a node from a BST.

I MCA I SEM Statistical Programming with R Lab – MC1718

1. Write a program to illustrate basic Arithmetic in R
2. Write a program to illustrate Variable assignment in R
3. Write a program to illustrate data types in R
4. Write a program to illustrate creating and naming a vector in R
5. Write a program to illustrate create a matrix and naming matrix in R
6. Write a program to illustrate Add column and Add a Row in Matrix in R
7. Write a program to illustrate Selection of elements in Matrixes in R
8. Write a program to illustrate Performing Arithmetic of Matrices
9. Write a program to illustrate Factors in R
10. Case study of why you need use a Factor in R
11. Write a program to illustrate Ordered Factors in R
12. Write a program to illustrate Data Frame Selection of elements in a Data frame
13. Write a program to illustrate Sorting a Data frame
14. Write a program to illustrate List ? Why would you need a List
15. Write a program to illustrate Adding more elements into a List
16. Write a program to illustrate if-else-else if in R

17. Write a Program to illustrate While and For loops in R

18. Write a program to illustrate Compare and Matrices and Compare vectors

19. Write a program to illustrate Logical & and Logical | operators in R.

20. Write a program to illustrate Functions in Quick sort implementation in R

21. Write a program to illustrate Function inside function in R

22. Write a program to illustrate to create graphs and usage of plot() function in R

23. Write a program to illustrate Customizing and Saving to Graphs in R.

24. Write a program to illustrate some built in Mathematical Functions

I MCA II SEM **Subject Code: MC1721** **Subject Name: OOPS Through JAVA**

Course Objectives:

1) Students will learn how to write programs in JAVA

2) Students will learn different oops concepts in JAVA

3) Students will learn the various JAVA concepts like Applets, AWT, swings etc.

Course Prerequisites:

Students should have knowledge of

1) Basic Computer and programming knowledge.
2) Basic knowledge in Object Oriented Programming

Course Outcome:

Students will be able to:

1) understand the basic JAVA programming
2) knowledge on Inheritance, Packages and Interfaces in JAVA
3) knowledge on Exception Handling, Multi Threading in JAVA
4) Illustrate the concept of event handling mechanism in JAVA
5) Constructs window based application in JAVA

UNIT-I **OOP AND JAVA BASICS**

Basics of Object Oriented Programming(OOP): Need for OO paradigm , A way of viewing world- Agents, responsibility, messages, methods, classes and instances, class hierarchies(Inheritance), method binding, overriding and exceptions, summary of oop concepts, coping with complexity, abstraction mechanisms

Java Basics: Data types, variables, scope and life time of variables, arrays, operators, expressions, control statements, type conversion and costing, simple java program, classes and objects- concepts of classes, objects, constructors, methods, access control, this keyword, garbage collection, overloading methods and constructors, parameter passing, recursion, string handling.

UNIT-II **INHERITANCE, PACKAGES AND INTERFACES**

Inheritance: Hierarchical abstractions, Base class object, subclass, subtype, substitutability, forms of inheritance- specialization, specification, construction, extension, limitation, Combination, benefits of inheritance costs of inheritance. Member access rules, super uses, using final with inheritance, polymorphism, abstract classes.

Packages and Interfaces: Defining, Creating and Accessing a package, Understanding CLASSPATH, Importing packages, differences between classes and interfaces, defining an interface, Implementing interface, applying interfaces variables in interface and extending interfaces.

UNIT-III **EXCEPTION HANDLING AND MULTITHREADING**

Exception handling and Multithreading: Concepts of exception handling, benefits of exception handling, Termination or presumptive models, exception hierarchy, usage of try, catch, throws and finally, built in exceptions, creating own exception sub classes. Differences between multi threading and multitasking, thread life cycle, creating threads, synchronizing threads, daemon threads, thread groups

UNIT-IV **EVENT HADNLING**

Event Handling: Events, Event sources, Event classes, Event Listeners, Delegation event model, handling mouse and keyboard events, Adapter classes, inner classes. The AWT class hierarchy , user-interface components- labels, button, canvas, scrollbars, text components, check box, check box groups, choices, list panes- scroll pane, dialogs, menu bar, graphics, layout manager- layout manager types- boarder, grid, flow, card and grid bag.

UNIT-V **APPLETS & SWINGS**

Applets: Concepts of Applets, differences between applets and applications, lifecycle of an applet, types of applets, creating applets, passing parameters to applets.

Swings: Introduction, limitations of AWT, MVC architecture, components, containers, exploring swing- JApplet, JFrame and JComponent, Icons and Labels, text fields, buttons-The JButton class, Check boxes, Radio Buttons, Combo boxes, Tabbed panes, Scroll panes, Trees and Tables

Text Books:

1) Java-The complete reference,7/e, Herbert Scheldt, TMH.

2) JAVA: How to program, 8/e, Dietal, Dietal, PHI.

3) 3.1.1 Introduction of programming with JAVA, S. Dean, TMH.

4) Introduction to Java programming, 6/e, Y. Daniel Liang, Pearson.

Reference Books:

1) Core Java 2, Vol. 1(Vol. 2) Fundamentals (Advanced), 7/e, Cay. S. Horstmann, Gary Cornell, Pearson.

2) Big Java2, 3/e, Cay. S. Horstmann, Wiley.

3) Object Oriented Programming through Java, P. Radha Krishna, University Press.

4) JAVA& Object Orientation an Introduction, 2/e, John Hunt, Springer

5) Introduction to JAVA Programming, 7/e, Y. Daniel Liang, Pearson. , TMH.

I MCA II SEM Subject Code: MC1722 Subject Name: Operating System

Course Objectives:

1) Students will learn the basics of designing operating system and they can for designing the new OS and can learn from experiences.

2) Students will learn different type of process management along with CPU scheduling and resource utilization.

3) Students will learn the key concept such as how to do memory management and handling of deadlocks.

Course Prerequisites:

Students should have knowledge of

1) Basic structure of OS and its design process

2) Basic process and memory management

3) Security policy and deadlock management

Course Outcome:

Students will be able to:

1. Explain the different types of operating system and its architecture like distributed system, special purpose system and the working principle of system calls with Virtual machine.
2. Classifying the process management with typical concept of waiting time and turnaround time of different CPU scheduling algorithms. Describing the need and synchronization of process and resource utilization with preventive measure of conflict.
3. Classifying the different type of memory management scheme and examine the page faults for different Page Replacement algorithms and allocation of frames.
4. Define the different characteristics of deadlock occurrence and corrective measure for recovery. Explain the security and privacy issues measurement for the data and system.
5. Describing the file management scheme and its implementation, explaining different type of file organization and allocation method and at last structure of disk scheduling and its structure.

UNIT-I

Introduction to System

Computer system organization, Computer- system Architecture, Operating system Structure, Operating-system Operations, Process Management, Memory Management, Storage management, Protection and Security, Distributed Systems, Special-purpose systems, Computing Environments, Operating-system Services, Operating-system Interface, System calls, Operating-system Design and Implementation, Operating-system structure, Virtual Machine.

UNIT-II

Process Management

Process Concept, Process Scheduling, IPC and examples of IPC Systems, Threads: Overview, Threading Issues, Multithreading Models, CPU Scheduling: Basic concepts, Scheduling Criteria, Scheduling Algorithms, Multiple-Processor Scheduling, Thread Scheduling, Process Synchronization: The Critical- section problem, Petersons solution, Synchronization Hardware, Semaphores, Classic problems of Synchronization, Monitors, Atomic Transactions.

UNIT-III

Memory management

Main memory: Swapping, Contiguous memory Allocation, Paging, Structure of the Page table, Segmentation Virtual memory: Background, Demand paging, copy on-Write, Page Replacement, Allocation of frames, Thrashing, Memory-Mapped Files.

UNIT-IV

Deadlocks and Security

System model, Deadlock Characterization, Methods for handling Deadlocks, Deadlock Prevention, Deadlock avoidance Deadlock Detection and Recovery form Deadlock. Protection: Goals of Protection, Principles of protection, Access Matrix, Access Control, Revocation of Access Rights, Capability Based systems, Language-Based Protection.

The Security Problem, Program Threads, System and Network Threats, Cryptography as a security tool, User Authentication, Implementing security Defenses, Firewalling to protect systems and Networks.

UNIT-V

File-system Implementation

File-system Structure, Implementation, Directory Implementation, Allocation Methods, Free- Space Management, Efficiency and Performance, Recovery, Log-Structured File systems, NFS Mass –storage Structure: Overview, Disk Structure, Disk Attachment, Disk Scheduling, Disk and swap-space Management, RAID Structure, Stable- Storage Implementation, Tertiary-Storage Structure I/O systems.

Text Books:

1) Operating system concepts, 7/e, Abraham Siliberschatz, Galvin, John Wiley &sons ,

Reference Books:

1) Operating systems, 6/E, William stallings, PHI/Pearson.

2) Operating systems 3/e, Dietal, Dietal, Pearson.

3) Operating systems, 2/e, Dhamdhere, TMH.

4) An introduction to Operating systems, Concepts and practice, Pramod Chandra P. Bhat, PHI

I MCA II SEM Subject Code: MC1723 Subject Name :Software Engineering

Course Objectives:

1) The students will know about the various myths that exist in the software industry
2) The meaning of a process and the various process models
3) The various methods of Requirements elicitation, prioritization, specification and validation of requirements
4) The meaning of architecture and various architectural styles
5) The principles and guidelines for good coding
6) The testing process, various testing strategies and testing techniques

Course Prerequisites:

Students should have knowledge of

1) Basics of software and programming **knowledge**

Course Outcome:

Students will be able to:

1) The students will know about the various myths that exist in the software industry
2) The meaning of a process and the various process models
3) The various methods of Requirements elicitation, prioritization, specification and validation of requirements
4) The meaning of architecture and various architectural styles
5) The principles and guidelines for good coding
6) The testing process, various testing strategies and testing techniques

UNIT-I **Introduction to Software Engineering**

The evolving role of software, Changing Nature of Software, Software myths.
(Text Book 3)

The software problem: Cost, schedule and quality, Scale and change

UNIT-II **Software Process**

Process and project, component software process, Software development process models : Waterfall model, prototyping, iterative development, relational unified process, time boxing model, Extreme programming and agile process, using process models in a project. Project management process

UNIT-III **Software requirement analysis and specification**

Value of good SRS, requirement process, requirement specification, functional specifications with use-cases, other approaches for analysis, validation.

Planning a software project: Effort estimation, project schedule and staffing, quality planning, risk management planning, project monitoring plan, detailed scheduling.

UNIT-IV **Software Architecture**

Role of software architecture, architecture views, components and connector view, architecture styles for C & C view, documenting architecture design, evaluating architectures.

Design: Design concepts, function-oriented design, object oriented design, detailed design, verification, metrics

UNIT-V **Coding and Unit testing**

Programming principles and guidelines, incrementally developing code, managing evolving code, unit testing, code inspection and metrics.
Testing: Testing concepts, testing process, black-box testing, white-box testing, metrics.

Text Books:

1) A Concise introduction to software engineering (undergraduate topics in computer science), Pankaj Jalote, Springer International Edition.
2) Software Engineering, A Precise approach, Pankaj Jalote, Wiley
3) Software Engineering, 3/e ,& 7e Roger S.Pressman , TMH

Reference Books:

1) Software Engineering, 8/e, Sommerville, Pearson.

2) Software Engineering principles and practice, W S Jawadekar, TMH
3) Software Engineering concepts, R Fairley, TMH

I MCA II SEM: OPERATIONS RESEARCH FOR COMPUTER APPLICATIONS: MC1724

UNIT–I:

Development: Definition, Characteristics and Phases, scientific method. Types of models, general methods for solving, operations research modes.

Allocation: introduction linear programming formulation, graphical solution, simplex methods, artificial variable technique, duality principle.

UNIT-II:

Transportation problem: Formulation, optimal solution, unbalanced transportation, assignment problem: formulation, optimal solution, Travelling Salesman Problem.

UNIT-III: Replacement: Introduction, replacement of items that deteriorate when money value is not counted and counted, and replacement of items that fail completely (i.e.) group replacements.

Waiting lines: Introduction , Queueing system, poisson arrivals, exponential service, Single server Queueing models.

UNIT-IV: Inventory: Introduction, single item, deterministic models, production is instantaneous or at a constant rate , shortages are allowed or not allowed and with drawls from stock is continuous, purchase inventory model with one price break ,shortages are not allowed.

UNIT–V:

Game Theory: Introduction – Two Person Zero-Sum Games, Pure Strategies, Games with Saddle Point, Mixed strategies, Rules of Dominance, Solution Methods of Games without Saddle point – Algebraic, matrix and arithmetic methods.

Project Management: PERT and CPM , difference between PERT and CPM, PERT/CPM network components and precedence relations, Time Estimates for activities.

TEXTBOOKS:

1. Operations Research, S.D.Sharma, Ramnath, & Kedarnath co, Meerut.

REFERENCES:

1. Operations Research, P.K.Gupta, D.S. Hira, S.Chand.
2. Operations Research, R.D.Asrhedkar, R.V.Kulkarni.
3. Operations Research, Problems & sollutons, 3/e, JKSharma, Macmillan.
4. Operations Research, 8/e, Hillier, Liberman, TMH.
5. Operations Research, An introduction , 8/e, Taha, Pearson.

I MCA II SEM: COMPUTER GRAPHICS: MC1725

Course Objectives:

1. To develop an ability to analyze the requirements of the software, understand the technical specifications, design and provide novel engineering solutions and efficient product designs
2. Students will learn different algorithms for designing line, circle, ellipse, area filling methods
3. Students will learn the concepts of computer graphics 2D,3D,Animation

Course Prerequisites: Students should have knowledge of

1) Basic knowledge on Computer Graphics
2) Basic knowledge in Mathematics
3) Basic Computer and programming knowledge.

Course Outcome:

1) understand the basic s of Computer Graphics
2) knowledge on simulators,2D,3D,Animation
3) knowledge on object representation, line, circle Design
4) Awareness on programs of computer graphics

UNIT-I: Introduction

Introduction: Application areas of Computer Graphics, overview of graphics systems, video display devices, raster-scan systems, random scan systems, graphics monitors and work stations. Input devices, Input interaction, Display lists, Menus, picking.
Output primitives : Points and lines, line drawing algorithms, mid-point circle and ellipse algorithms. Filled area primitives: Scan line polygon fill algorithm, boundary-fill and flood-fill algorithms.

UNIT-II: 2-D geometrical transformations, 2-D viewing

2-D geometrical transforms: Translation, scaling, rotation, reflection and shear transformations, matrix representations and homogeneous coordinates, composite transforms, between coordinate systems.

2-D viewing: The viewing pipeline, viewing coordinate reference frame, window to view-port coordinate transformation, viewing functions, Cohen- Sutherland and Cyrus-beck line clipping algorithms, Sutherland –Hodgeman polygon clipping algorithm.

UNIT-III: 3-D object representation,3-D Geometric transformations

3-D object representation: Polygon surfaces, quadric surfaces, spline representation, Hermit curve, Bezier curve and B-Spline curves, Bezier and B-Spline surfaces. Basic illumination models, polygon rendering methods.
3-D Geometric transformations: Translation, rotation, scaling, reflection and shear transformations, composite transformations.

UNIT-IV: 3-D Viewing, Visible surface Detection methods

3-D viewing: Viewing pipeline, viewing coordinates, view volume and general projection transforms and clipping.

Visible surface detection methods: Classification, back-face detection, depth-buffer, scan-line, depth sorting, BSP-tree methods, area sub-division and octree methods.
C Graphics Basics: Graphics programming, initializing the graphics, C Graphical functions, simple programs

UNIT-V: Computer Animation

Computer animation: Design of animation sequence, general computer animation functions, raster animation, computer animation languages, key frame systems, motion specifications.

Text Books:

1) Computer Graphics *C version*", Donald Hearn and M. Pauline Baker, Pearson Education.
2) "Computer Graphics Principles & practice", second edition in C

Reference Books:

1) "Computer Graphics", second Edition, Donald Hearn and M. Pauline Baker, PHI/Pearson Education.
2) Procedural elements for Computer Graphics, David F Rogers, Tata Mc Graw hill, 2nd edition.
3) Principles of Interactive Computer Graphics", Neuman and Sproul, TMH

I MCA II SEM: OOPS THROUGH JAVA LAB – MC1726

Course Objectives:

• To learn/strengthen a programming language JAVA, To learn problem solving techniques

• To introduce the student to java concepts such as Exception handling, applets, swings etc.,

Course Outcomes:

- Able to write programs in Java Language
- Develop logical and analytical thinking in programming concepts.
- Knowledge in writing programs in various concepts like Exception Handling, applets, swings etc.

□ How to read and write contents from or into a file Use JDK 1.5 or above on any platform e.g. Windows or UNIX.

□ Student is expected to complete any 15 programs.

1. The Fibonacci sequence is defined by the following rule. The first 2 values in the sequence are 1, 1.every subsequent value is the sum of the 2 values preceding it. Write A Java Program (WAJP) that uses both recursive and non recursive functions to print the nth value of the Fibonacci sequence.

2. WAJP to demonstrate wrapper classes and to fix the precision.

3. WAJP that prompts the user for an integer and then prints out all the prime numbers

up to that Integer.

4. WAJP that checks whether a given string is a palindrome or not. Ex. MALAYALAM is a palindrome.

5. WAJP for sorting a given list of names in ascending order.

6.WAJP to check the compatibility for multiplication, if compatible multiply two matrices and find its transpose.

7. WAJP that illustrates how runtime polymorphism is achieved.

8. WAJP to create and demonstrate packages.

9.WAJP, using String Tokenizer class, which reads a line of integers and then displays each integer and the sum of all integers.

10. WAJP that reads on file name form the user then displays information about whether the file exists, whether the file is readable/writable, the type of file and the length of the file in bytes and display the content of the using FileInputStream class.

11. WAJP that displays the number of characters, lines and words in a text/text file.

12. Write an Applet that displays the content of a file.

13. WAJP that works as a simple calculator. Use a grid layout to arrange buttons for the digits and for the +-*?% operations. Add a text field to display the result.

14. WAJP for handling mouse events.

15. WAJP demonstrating the life cycle of a thread.

16. WAJP that correctly implements Producer-Consumer problem using the concept of Inter Thread Communication.

17. WAJP that lets users create Pie charts. Design your own user interface(with Swings & AWT).

18. WAJP that allows user to draw lines, rectangles and ovals.

19. WAJP that implements a simple client/server application. The client sends data to a server. The server receives the data, uses it to produce a result and then sends the result back to the client. The client displays the result on the console. For ex: The data send form the client is the radius of a circle and the result produced by the server is the area of the circle.

20. WAJP to generate a set of random numbers between two numbers x1 and x2, and x1>0.

21. WAJP to create an abstract class named shape, that contains an empty method named number Of Sides(). Provide three classes named Trapezoid, Triangle and Hexagon, such that each one of the classes contains only the method number Of Sides(), that contains the number of sides in the given geometrical figure.

22. WAJP to implement a Queue, using user defined Exception Handling (also make use of throw, throws).

23. WAJP that creates 3 threads by extending Thread class. First thread displays “Good Morning” every 1 sec, the second thread displays “Hello” every 2 seconds and the third displays “Welcome” every 3 seconds. (Repeat the same by implementing Runnable).

24. Create an inheritance hierarchy of Rodent, Mouse, Gerbil, Hamsteretc. In the base class provide methods that are common to all Rodents and override these in the derived classes to perform different behaviours, depending on the specific type of Rodent. Create an array of Rodent, fill it with different specific types of Rodents and call your base class methods.

I MCA II SEM: Web Technologies Lab: MC1727

Course Objectives

The objectives of the course are:

- Understand the importance of the web as a medium of communication
- Understand the principles of creating an effective web page, including an in-depth consideration of information architecture..
- Become familiar with graphic design principles that relate to web design and learn how to implement these theories into practice
- Develop skills in analyzing the usability of a web site.
- Learn the language of the web: HTML and CSS.

Course Outcomes

After completing this Course, the student should be able to:

- Apply critical thinking and problem solving skills required to successfully design and implement a web site.
- Demonstrate the ability to analyze, identify and define the technology required to build and implement a web site
- Demonstrate knowledge of artistic and design components that are used in the creation of a web site.
- Utilize and apply the technical, ethical and interpersonal skills needed to function in a cooperative environment.

List of Programs

1. Create a HTML page and display text on the page using font styles and fore ground and background colors.
2. Create a Simple and Complex tables using images and data.
3. Create a page with hyperlink and frames concept.
4. Create a form using different controls available.
5. Create a page using Inline style sheets.
6. Create a page using embedded style sheets.
7. Create a page using Linked external style sheets.
8. Create a page using mouse functions.
9. Create a page for random number generation logic (Dice)
10. Create a page which reads input values and perform arithmetic operations using java script.
11. Design a calculator using JavaScript.
12. Implement date and time functions using java script.

I MCA II SEM: Operating Systems & Computer Graphics Lab: MC1728

1. Implementation of DDA Line Algorithm

2. Implementation of Bresenham's Line Algorithm

3. Implementation of Midpoint Circle Algorithm

4. Implementation of Midpoint Ellipse Algorithm

5. Implementation of Two Dimensional Transformations

6. Implementation of Two Dimensional Composite Transformations

7. Simulate the Following CPU Scheduling Algorithms A) Round Robin B) Sjf C) Fcfs D) Priority

8. Multiprogramming-Memory Management- Implementation Of Fork (), Wait (), Exec () And Exit ()

9. Simulate The Following
    a. Multiprogramming with A Fixed Number Of Tasks (Mft)
    b. Multiprogramming with A Variable Number Of Tasks (Mvt)

12. Simulate Bankers Algorithm for Dead Lock Avoidance

13. Simulate Bankers Algorithm for Dead Lock Prevention.

14. Simulate The Following Page Replacement Algorithms.

    A) Fifo B) Lru C) Lfu

15. Simulate the Following File Allocation Strategies
    A) Sequenced B) Indexed C) Linked

II MCA DETAILED SYLLABUS
(3rd & 4th Semesters)

Subject Code	DATABASE MANAGEMENT SYSTEMS	L	T	P	Credits
4010172301		04	01	0	04

Course Overview:

This course introduces database design and creation using a DBMS product. Emphasis is on data dictionaries, normalization, data integrity, data modeling, and creation of simple tables, queries, reports, and forms. Upon completion, students should be able to design and implement normalized database structures by creating simple database tables, queries, reports, and forms.

Course Objectives:

1. Provide students with theoretical knowledge and practical skills in the use of database and database management systems in information technology applications.
2. The logical design, physical design and implementation of relational databases are covered.

Course Outcomes:

1. Student can able to describe the Architecture of Database Management Systems
2. Student can design different ER Models
3. Student can able to differentiate the knowledge in TRC & DRC
4. Student can compare relational model with the structured query language (SQL)
5. Student can able to design the new database
6. Student can perform transactions for new concepts
7. Student can differentiate different indexing techniques in real time

Unit-I:

Introduction to Database & ER Diagrams

Introduction to Database:
Database System Applications, Purpose of Database Systems, View of Data – Data Abstraction, Instances and Schemas, Data Models – the ER Model, Relational Model, Other Models – Database Languages –DDL, DML, Database Access from Applications Programs, Transaction Management, Data Storage and Querying, Database Architecture, Database Users and Administrators, History of Data base Systems. Introduction to Data base design,

ER diagrams:
Introduction, Beyond ER Design, Entities, Attributes and Entity sets, Relationships and Relationship sets, Additional features of ER Model, Conceptual Design with the ER Model, Conceptual Design for Large enterprises, Logical Database Design.

Outcome:

1. Student can able to describe the Architecture of Database Management Systems.
2. Student can design different ER Models.

Activity/Event :

Seminar and Class Test.

Unit-II:

Relational Algebra and Calculus & Relational Model

Relational Algebra and Calculus: Relational Algebra – Selection and Projection, Set operations, Renaming, Joins, Division, Examples of Algebra Queries, Relational calculus – Tuple relational Calculus – Domain relational calculus

Relational Model: Introduction to the Relational Model – Integrity Constraints Over Relations, Enforcing Integrity constraints, Introduction to Views – Destroying /altering Tables and Views.

Outcome:

1. Student can able to differentiate the knowledge in TRC & DRC
2. Student can compare relational model with the structured query language (SQL)

Activity/Event :

Seminar and Class Test.

Unit-III:

SQL & Schema Refinement

SQL:

Form of Basic SQL Query – Examples of Basic SQL Queries, Introduction to Nested Queries, Correlated Nested Queries, Set – Comparison Operators, Aggregate Operators, NULL values – Comparison using Null values – Logical connectives – AND, OR and NOT – Impact on SQL Constructs, Outer Joins, Disallowing NULL values, Complex Integrity Constraints in SQL Triggers.

Introduction to Schema Refinement :

Problems Caused by redundancy, Decompositions – Problem related to decomposition, Functional Dependencies - Reasoning about FDS, Normal Forms – FIRST, SECOND, THIRD Normal forms – BCNF –Properties of Decompositions- Loss less join Decomposition, Dependency preserving Decomposition, Schema Refinement in Data base Design – Multi valued Dependencies – FOURTH Normal Form, Join Dependencies, FIFTH Normal form, Inclusion Dependencies.

Outcome:

1. Student can able to design the new database

Activity/Event :

Seminar and Class Test

Unit-IV:

Transaction Management & Crash Recovery:

Overview of Transaction Management:

The ACID Properties ,Concurrent execution of Transactions.Concurrency Control Techniques : Scheduling , Serializability ,Basic concepts on Locking Techniques ,Lock Based Concurrency Control ,Deadlocks -- Performance of Locking , Concurrency Control without Locking

Crash recovery:

Introduction to Crash recovery, Introduction to ARIES ,the Log and Other Recovery related Structures ,the Write-Ahead Log Protocol ,Check pointing ,recovering from a System Crash ,Media recovery.

Outcome:

1. Student can perform transactions for new concepts

Activity/Event :

Seminar and Class Test

Unit-V:

Storage & Indexing

Overview of Storage and Indexing:

Data on External Storage, File Organization and Indexing – Clustered Indexes, Primary and Secondary Indexes, Index data Structures – Hash Based Indexing, Tree based Indexing. Storing data: Disks and Files: -The Memory Hierarchy – Redundant Arrays of Independent disks. Tree Structured Indexing: Intuitions for tree Indexes, Indexed Sequential Access Methods (ISAM) B+ Trees: A Dynamic Index Structure, Search, Insert and Delete. Hash Based Indexing: Static Hashing, Extendable hashing, Linear Hashing, Extendible vs. Linear Hashing.

Outcome:

1. Student can differentiate different indexing techniques in real time

Activity/Event :

Seminar and Class Test

Text Books:

1. Database Systems, A Practical approach to Design Implementation and Management Fourth edition, Thomas Connolly, Carolyn Begg, Pearson education.
2. Data base System Concepts, A. Silberschatz, H.F. Korth, S. Sudarshan, McGraw hill, VI edition, 2006.
3. Fundamentals of Database Systems 5th edition, Ramez Elmasri, Shamkant .Navathe, Pearson Education,2008.

Reference Books:

1. Database Systems, A Practical approach to Design Implementation and Management Fourth edition, Thomas Connolly, Carolyn Begg, Pearson education.
2. Database Management System Oracle SQL and PL/SQL, P.K. Das Gupta, PHI.
3. Introduction to Database Management, M. L. Gillenson and others, Wiley Student Edition.
4. Introduction to Database Systems, C. J. Date, Pearson Education.

Subject Code	ADVANCED JAVA PROGRAMMING	L	T	P	Credits
4010172302		04	01	0	04

Course Overview:

This course of study builds on the skills gained by students in Advanced Java concepts and helps to advance Java programming skills. Students will design object-oriented applications with Java and will create Java programs using hands-on, engaging activities

Course Objectives:

1. To understand the concepts of Hypertext Markup Language and Cascading Style Sheets.
2. To learn JavaScript for creating dynamic websites.
3. To learn the operations perform on data among web applications using XML
4. To acquire knowledge on creation of software components using JAVA Beans.
5. To learn Server-Side Programming using Servlets and Java Server Pages.
6. To learn the creation of pure Dynamic Web Application using JDBC.

Course Outcomes:

1. Students are able to develop a dynamic webpage by the use of java script and DHTML.
2. Students will be able to connect a java program to a DBMS and perform insert, update and delete operations on DBMS table.
3. Students will be able to write a server side java application called Servlet to catch form data sent from client, process it and store it on database.
4. Students will be able to write a server side java application called JSP to catch form data sent from client and store it on database
5. Students are able to develop a dynamic webpage by the use of java script and DHTML.

Unit-I:

Review of HTML4:

Common tags, HTML Tables and formatting internal linking, Complex HTML forms. Introduction to Scripting Languages: Java Scripts, Control structures, functions, arrays & objects, DHTML, CSS, event model, filters & transitions.

Outcome:

Outline the origin of markup languages and develop basic web page/s.
Create a link from one webpage to another and apply image insertion into web page/s and also make hyperlinks to images.
Create, modify and format the contents of web page with CSS.
Create dynamic interactive Web pages using JavaScript.
Use Java Script to validate form entries.
Create HTML that can change even after a page has been loaded into a browser.

Activity/Event :

Seminar and Class Test

Unit-II:

Review of Applets, Class, Event Handling, AWT Programming,

Introduction to Swing:

JApplet, Handling Swing Controls like Icons, Buttons, Text Boxes,
Combo Boxes, Tabbed Panes, Scroll Panes, Trees, Tables, Differences between AWT Controls & Swing Controls, Developing a Home page using Applets & Swing

Outcome:

Capability to develop applications through CORE JAVA features like: Events, Exceptions, built-in java objects, Applets and Frames, AWT and Swings.

Activity/Event :

Seminar and Class Test

Unit-III:

Java Beans, Introduction to Serve lets:

Introduction to Java Beans, Advantages of Java Beans, BDK, Introspection, Using Bound properties, Bean Info Interface, Constrained properties, Persistence, Customizers, Java Beans API.

Lifecycle of a Servlet, JSDK, The Servlet API, The javax.servlet Package, Reading Servlet parameters, Reading Initialization Parameters, The javax.Servlet.HTTP package, Handling, Http Request & responses, Using Cookies, Session Tracking, Security Issues.

Outcome:

- Outline the fundamental principles and methodologies of constructing component software including component standards, architectures, interface, implementations, and integrations.
- Design components to solve a problem, and evaluate alternatives.
- Utilize effectively the software component development tools to develop software components such as Java Beans.
- Analyze the role of Java Servlets in Java 2 Enterprise Edition architecture and as the best Java solution to HTTP application development.
- Use request and response objects provided to a servlet to read parameters and to produce an HTML response.
- Develop interactive web applications using HTML forms and servlets.
- Demonstrate the complex conversation with HTTP clients using session attributes.

Activity/Event :

Seminar and Class Test

Unit-IV:

Introduction to JSP, Setting Up the JSP Environment, JSP Application Development:

The Problem with Serve lets, The Anatomy of a JSP Page, JSP Processing, JSP Application Design with MVC.
Installing the Java Software Development Kit, Tomcat Server & Testing Tomcat.
Generating Dynamic Content, Using Scripting Elements, Implicit JSP Objects, Conditional Processing – Displaying Values, Using an Expression to Set an Attribute, Declaring Variables and Methods, Error Handling and Debugging, Sharing Data Between JSP Pages, Requests, and Users, Passing Control and Data Between Pages – Sharing
Session and Application Data Memory Usage Considerations.

Outcome:

- Explain the JSP technology, its features and advantages.
- Discuss the architecture of web-based systems.
- Describe Web development process and varies server-side technologies.
- Develop JSP Applications with JSP tags, JSP scriptlets and java beans.
- Distinguish JSP Application Models.
- Develop JSP applications implementing Session¬ management and Data base Connectivity.

Activity/Event :

Seminar and Class Test.

Unit-V:

Database Access:

Database Programming using JDBC, Studying Javax.sql. * package. Accessing a database from a JSP Page, Application – Specific Database Actions, Deploying JAVA Beans in a JSP Page.

Outcome:

- Outline the fundamentals of JDBC and its importance, uses, strengths and weaknesses.
- Explain the JDBC architecture and required components.
- Manage normal and abnormal interactions with databases using JDBC.
- Evaluate the process results using scrollable result sets and row sets.

Activity/Event :

Seminar and Class Test

Text Books:

1. Internet and World Wide Web: How to program, 6/e, Dietel, Dietel, Pearson.
2. The Complete Reference Java2, 8/e, Patrick Naughton, Herbert Schildt, TMH.
3. Java Server Faces, Hans Bergstan, O'reilly.

Reference Books:

1. Web Programming, building internet applications, 2/e, Chris Bates, Wiley Dreamtech
2. Programming World Wide Web, Sebesta, PEA
3. Web Technologies, 2/e, Godbole, kahate, TMH
4. An Introduction to web Design , Programming ,Wang, Thomson

Subject Code	Unix Programming	L	T	P	Credits
4010172303		04	01	0	04

Course Overview:

The course presents the concepts necessary to understand the way UNIX works as well as the system's most commonly used commands. Data manipulation utilities and shell syntax for synthesizing command pipelines are emphasized. Bourne shell, Bash shell and Korn shell programming techniques are introduced so students will be able to read and modify existing shell scripts as well as create their own. Comprehensive hands on exercises are integrated throughout to reinforce learning and develop real competency

Course Objectives:

1. To understand the fundamental design of the unix operating system.
2. To become fluent with the systems calls provided in the unix environment.
3. To be able to design and build an application/service over the unix operating system

Course Outcomes:

1. Ability to understand and reason out the working of Unix Systems
2. To teach students the use of basic UNIX Utilities
3. To teach students the principles of UNIX shell programming
4. To familiarize students with the concepts, design, and structure of the UNIX operating system
5. To be able to build an application / service over a UNIX system

Unit-I:

Review of Unix Utilities and Shell Programming

File handling utilities, security by file permissions, process utilities, disk utilities, networking commands, backup utilities, text processing utilities, Working with the Bourne shell-, What is a shell, shell responsibilities, pipes and input redirection, output redirection, here documents, the shell as a programming language, shell meta characters, shell variables, shell commands, the environment, control structures, shell script examples.

Outcome:

1. Ability to understand and reason out the working of Unix Systems

Activity/Event :

Seminar and Class test

Unit-II:

Unix Files

Unix file structure, directories, files and devices, System calls, library functions, low level file access, usage of open, create, read, write, close, lseek, stat, fstat, octl, umask, dup, dup2. The standard I/O (fopen, fclose, fflush, fseek, fgetc, getc, getchar, fputc, putc, putchar, fgets, gets), formatted I/O, stream errors, streams and file descriptors, file and directory maintenance (chmod, chown, unlink, link, symlink, mkdir, rmdir, chdir, getcwd), Directory handling system calls (opendir, readdir, closedir, rewinddir, seekdir, telldir)

Outcome:

1. To teach students the use of basic UNIX Utilities

Activity/Event :

Seminar and Class test

Unit-III:

Unix Process

Threads and Signals: What is process, process structure, starting new process,waiting for a process, zombie process, process control, process identifiers, system call interface for process management, -fork, vfork, exit, wait, waitpid, exec, system, Threads, -Thread creation, waiting for a thread to terminate, thread synchronization, condition variables,cancelling a thread, threads vs. processes, Signals-, Signal functions, unreliable signals,interrupted system calls, kill and raise functions, alarm, pause functions, abort.

Outcome:

1. To teach students the principles of UNIX shell programming

Activity/Event :

Seminar and Class test

Unit-IV:

Data Management & Message Queues

Data Management:

Management Memory (simple memory allocation, freeing memory) file and record locking (creating lock files, locking regions, use of read/ write locking, competing locks, other commands, deadlocks). Inter process Communication: Introduction to IPC, IPC between processes on a single computer system, IPC between processes on different systems, pipes, FIFOs, streams and messages, namespaces, introduction to three types of IPC (system-V)-message queues, semaphores and shared memory

Message Queues-:

IPC, permission issues, Access permission modes, message structure, working message queues, Unix system-V messages, Unix kernel support for messages, Unix APIs for messages, client/server example.

Outcome:

1. To familiarize students with the concepts, design, and structure of the UNIX operating system

Activity/Event :

Seminar and Class test

Unit-V:

Semaphores & Sockets

Semaphores: -

Unix system-V semaphores, Unix kernel support for semaphores, Unix APIs for semaphores, file locking with semaphores. Shared Memory: -Unix system-V shared memory,working with a shared memory segment, Unix kernel support for shared memory, Unix APIs for shared memory, semaphore and shared memory example.

Sockets:

Berkeley sockets, socket system calls for connection oriented protocol and connectionless protocol, example- client/server program,advanced socket system calls, socket options.

Outcome:

1. To be able to build an application / service over a unix system

Activity/Event : Seminar and Class test

Text Books:

1. Unix and shell Programming, N B Venkateswarlu, Reem
2. Unix Concepts and Applications, 3/e, Sumitabha Das, TMH

Reference Books:

1. Unix and shell Programming, Sumitabha Das, TMH
2. A Beginner's Guide to Unix, N.P.Gopalan, B.Sivaselva, PHI
3. Unix Shell Programming, Stephen G.Kochan, Patrick Wood, 3/e, Pearson
4. Unix Programming, Kumar Saurabh, Wiley,India
5. Unix Shell Programming, Lowell Jay Arthus & Ted Burns,3/e, GalGotia
6. Nix Concepts and Applications, Das, 4/e, TMH

Subject Code	MANAGEMENT INFORMATION SYSTEM	L	T	P	Credits
4010172304		4	1	0	4

Course Overview:

This course will introduce you to the concept of MIS and the impact it has on business organizations. Most people recognize that information systems are composed of technologies such as computers, keyboards, and networks, but technology is just one small component. Some argue that other components of MIS are far more important. Information systems are made up three high-level components: technology, people, and process

Course Objectives:

- To Define information technology and state the advancement of IT
- To Identify the different types of Information systems
- To Describe the evolution of MIS
- To Recognize the importance of having a disaster recovery plan
- To Explain the role of Computer Aided Decision System in business environment
- To Measure the threat of virus and identify ways of preventing them

Course Outcomes:

- The students will know to Define information technology and state the advancement of IT
- The students will know to Identify the different types of Information systems
- Describe the evolution of MIS
- To Recognize the importance of having a disaster recovery plan
- Understand the role of Computer Aided Decision System in business environment
- To Measure the threat of virus and identify ways of preventing them

Unit-I:

Management Information Systems

Management Information Systems: A Framework:

Importance of MIS, MIS: A Definition nature and Scope of MIS,
Structure and Classification of MIS: Structure of MIS, MIS Classification

Information and System Concepts:

Information: A Definition, Types of Information, Dimensions of Information, System: A Definition, Kinds of Systems, System Related Concepts, Elements of a System, Human as an Information Processing System

Information Systems for Competitive Advantage: Introduction, Changing concepts of Information System, Competitive Advantage, Information systems Strategies for Dealing with competitive Force, Porter's Value Chain Model, Strategic Information Systems (SIS)

Outcome:

- To understand why knowledge of information systems is important for business professionals,
- To Identify five areas of information systems knowledge

Activity/Event :

Seminar & Test

Unit-II:

Business Applications of IS

e-Commerce: Introduction, e – Commerce **ERP Systems:** Introduction, Enterprise Information Systems

Decision – Support Systems: Decision – Making: A Concept, Simon's Model of Decision – Making Types of Decisions, Methods for Choosing Among Alternatives, Decision – Making and MIS, Decision Support Systems – Why?, Decision Support Systems: A framework, Characteristics and Capabilities of DSS

Business Intelligence and knowledge Management System:

Business Intelligence, Knowledge Management System

Outcome:

- To enable understanding on enterprise information systems
- To enable knowledge on Knowledge Management Approach

Activity/Event :

Seminar & Test

Unit-III:

Information System Planning:

Information System Planning: Information System Planning: WHY? Planning Terminology Information System Planning, The Nolan Stage Model, The Four – Stage Model of IS planning Selecting A Methodology, Information Resources Management (IRM), Organization Structure and Location of MIS
System Acquisition: Acquisition of Information Systems, Acquisition of Hardware and Software

Outcome:

- To Understand the application of IT in the functional areas of management
- To gain knowledge on a Cross-functional enterprise system and its elements

Activity/Event :

Seminar & Test

Unit-IV:

System Implementation

System Implementation: Implementation Process, Organisational Change
Evaluation &Maintenance of IS: Evaluation of MIS, System Maintenance
IS Security and Control: IS Security Threats, Protecting Information System,
IS Security Technology The Disaster Recovery Plan

Outcome:

1. To understand the challenge of security threats faced by information systems (IS),
2. To enable the knowledge on the security management tools available for IS professionals

Activity/Event :

Seminar & Test.

Unit-V:

Building Of IS

System Development Approaches: System Development Stages, System Development Approaches

System Analysis and Design: System Analysis - Introduction, Requirement Determination, Strategies for Requirement Determination, Structured Analysis Tools

Systems Design: Design Objectives, Conceptual Design, Design Methods, Detailed System Design

Outcome:

- To enable understanding on system development approach
- To enable knowledge on system design

Activity/Event :

Seminar & Test

Text Books:

1. Management Information System, Managerial Perspectives, D P Goyal, 3 ed, McMillan Publications

Reference Books:

1. Management Information Systems Managing the Digital Firm, Kenneth C. , Jane P
2. Management information systems / James A. O'Brien, George M. Marakas O'Brien, James A.

Subject Code	DESIGN ANALYSIS OF ALGORITHMS	L	T	P	Credits
4010172305		04	01	0	04

Course Overview:

This course introduces basic methods for the design and analysis of efficient algorithms emphasizing methods useful in practice. Different algorithms for a given computational task are presented and their relative merits evaluated based on performance measures. The following important computational problems will be discussed: sorting, searching, elements of dynamic programming and greedy algorithms, advanced data structures, graph algorithms (shortest path, spanning trees, tree traversals), string matching, NP completeness.

Course Objectives:

1. Synthesize efficient algorithms in common engineering design situations.
2. Write rigorous correctness proofs for algorithms along with apply important algorithmic design paradigms and methods of analysis.
3. Demonstrate a familiarity with major algorithms and data structures.

Course Outcomes:

1. Basic data structure and it working topological design.
2. Basic functionality of different type of algorithms and its usage
3. Analysis of different type of complexity and its applicable condition

Unit-I:

Introduction to Algorithm & Design:

Algorithm, Pseudo code for expressing algorithms, performance Analysis-Space complexity, Time complexity, Asymptotic Notation- Big oh notation, Omega notation, Theta notation and Little oh notation, probabilistic analysis, Amortized analysis. Disjoint Sets- disjoint set operations, union and find algorithms, spanning trees, connected components and biconnected components.

Outcome:

Student will able to understand the complexity related to design algorithm

Activity/Event :

Seminar and Class Test

Unit-II:

Divide and conquer:

General method, applications-Binary search, Quick sort, Merge sort, Stassen's matrix multiplication. Greedy method: General method, applications-Job sequencing with deadlines, 0/1 knapsack problem, Minimum cost spanning trees, Single source shortest path problem.

Outcome:

1. Student Understand the knowledge in Relational Calculus
2. Student can compare relational model with the structured query language (SQL)

Activity/Event :

Seminar and Class Test

Unit-III:

Dynamic Programming:

General method, applications-Matrix chain multiplication, Optimal binary search trees, 0/1 knapsack problem, All pairs shortest path problem, Travelling sales person problem, Reliability design.

Outcome:

Student will able understand the application of branch and bound method with implementation.

Activity/Event :

Seminar and Class Test

Unit-IV:

Backtracking:

General method, applications-n-queen problem, sum of subsets problem, graph coloring, Hamiltonian cycles.

Outcome:

Will get an idea to implement the backtracking.

Activity/Event :

Seminar and Class Test.

Unit-V:

Branch and Bound:

General method, applications - Travelling sales person problem,0/1 knapsack problem- LC Branch and Bound solution, FIFO Branch and Bound solution. NP-Hard and NP Complete problems: Basic concepts, non deterministic algorithms, NP - Hard and NP Complete classes, Cook's theorem.

Outcome:

Student will get an idea for implementing a the diffent cases which will solve through branch & bound method.

Activity/Event :

Seminar and Class Test

Text Books:

1. Fundamentals of Computer Algorithms, Ellis Horowitz, Satraj Sahni and Rajasekharam, Universities Press.
2. Introduction to Algorithms, second edition, T. H. Cormen, C. E. Leiserson, R. L. Rivest and C. Stein, PHI Pvt. Ltd.

Reference Books:

1. Introduction to the Design and Analysis of Algorithms, Anany Levitin, PEA
2. Design and Analysis of Algorithms, Parag Himanshu Dave, Himansu Balachandra Dave, Pearson Education
3. Introduction to Design and Analysis of Algorithms A strategic approach, R.C.T. Lee, S. S. Tseng, R. C. Chang and T. Tsai, Mc GrawHill.

Subject Code	DATABASE MANAGEMENT SYSTEMS LAB	L	T	P	Credits
4010172321		0	0	3	2

Course Outcomes:

1. Have a good understanding of how several fundamental algorithms work, particularly those concerned with creation and updating of tables.
2 have a good understanding of the fundamental DBMS used in computer science
3 be able to understand various queries and their execution.
4 be able to design new database and modify existing ones for new applications and reason about the efficiency of the result

List of Experiments:

1. Execute a single line and group functions for a table.
2. Execute DCL and TCL Commands.
3. Create and manipulate various DB objects for a table.
4. Create views, partitions and locks for a particular DB.
5. Write PL/SQL procedure for an application using exception handling.
6. Write PL/SQL procedure for an application using cursors.
7. Write a DBMS program to prepare reports for an application using functions.
8. Write a PL/SQL block for transaction operations of a typical application using triggers.
9. Write a PL/SQL block for transaction operations of a typical application using package.
10. Design and develop an application using any front end and back end tool (make use of ER diagram and DFD).
11. Create table for various relation
12. Implement the query in sql for a) insertion b) retrieval c) updation d) deletion
13. Creating Views
14. Writing Assertion
15. Writing Triggers

16. Implementing operation on relation using PL/SQL

17. Creating Forms

18. Generating Reports

19. Typical Applications – Banking, Electricity Billing, Library Operation, Pay roll, Insurance, Inventory etc.

Text Books:

1. Database Management and Design by G.W. Hansen and J.V. Hansen, 2nd edition,1999, Prentice-Hall of India, Eastern Economy Edition.

Reference Books:

1. Database Management Systems by A.K. Majumdar and P. Bhattacharyya, 5 th edition, 1999, Tata McGraw-Hill Publishing.
2. A Guide to the SQL Standard, Date, C. and Darwen,H. 3rd edition, Reading, MA: 1994, AddisonWesley

Subject Code	UNIX Programming Lab	L	T	P	Credits
4010172322		0	0	3	2

Course Objectives:

This course introduces basic understanding of UNIX OS, UNIX commands and File system and to Familiarize students with the Linux environment To make student learn fundamentals of shell Scripting and shell programming Emphases are on making student familiar with UNIX environment and issues related to it.

Course Outcomes:

1. You will be able to run various UNIX commands on a standard UNIX/LINUX Operating
   System (We will be using Ubuntu flavor of the Linux operating system).
2. You will be able to run C / C++ programs on UNIX.
3. You will be able to do shell programming on UNIX OS.
4. You will be able to understand and handle UNIX system calls

List of Experiments

1. Program using basic network commands.
2. Program using system calls: create, open, read, write, close, stat, fstat, lseek .
3. Write a shell script for sorting, searching and insertion/deletion of elements in a list
4. Write a program to display the good morning, good afternoon, good evening and good night depending on the users log on time
5. Create two processes to run a for loop, which adds numbers 1 to n, say one process adds odd numbers and the other even all lines containing the word
6. Write a program to print prime numbers between x and y
7. Write a shell script which works similar to the wc command. This script can receive the option-l, -w, -c to indicate whether number of lines/ words/characters
8. Write a shell script that accepts any number of arguments and prints them in the reverse order.

9. Write a shell script that determines the period for which a specified user is working on the system
10. Program to implement inter process communication using pipes
11. Write a shell script that accepts a file name, starting and ending line numbers as arguments and displays all the lines between the given line numbers.
12. Write a shell script that deletes all lines containing a specified word in one or more files supplied as arguments to it.
13. Write a shell script that displays a list of all the files in the current directory to which the user has read,write and execute permissions.
14. Write a shell script that accepts a list of file names as its arguments, counts and reports the occurrence of each word that is present in the first argument file on other argument files.
15. Write a program which takes a set of filenames along with the command line and print thembased on their size in bytes either ascending or descending order
16. Write a shell script which displays a list of all files in the current directory to which you have read, write and execute permissions
17. Write a shell script for renaming each file in the directory such that it will have the current shell's PID as an extension. The shell script should ensure that the directories do not get renamed
18. Write a shell script to list all of the directory files in a directory
19. Write a shell script that copies multiple files to a directory.
20. Write a program demonstrating mutual exclusion principle
21. Write a program to demonstrate the locking mechanism while accessing the shared files Create a file that is shared among some users, write a program that finds whether a specificuser has created read and write operations on the file
22. Write a shell script that counts the number of lines and words present in a given file.
23. Write a program demonstrating semaphore operation on a shared file for reading but not Writing
24. Write a program which demonstrates the shared memory functions
25. Write a program which reads a source file name and destination file name using

command

26. line arguments and then converts into specified format (i.e. either from lower case to upper case or upper case to lower case or inverse of each)
27. Write a program to demonstrate the use of temporary files
28. Write a program which takes directory name along the command line and displays names ofthe files which are having more than one link
29. Write a program to demonstrate the use of exec family functions
30. Program using TCP sockets (Client and Server)
31. Program using UDP sockets (Client and Server)

Subject Code	Advanced JAVA Programming Lab	L	T	P	Credits
4010172323		0	0	3	2

Course Objectives:

1. HyperText Markup Language (HTML) and Cascading Style Sheets (CSS) for laying out (formatting) pages that contain text, images and graphics.
2. Extensible Markup Languages (XML is used to store and transport data among webpages), a mechanism for defining new tag sets and interchanging data among web applications.
3. Client-side Programming using JavaScript for validating the data.
4. Creation of software components (objects used for client and server communication) using Beans.
5. Server-Side Programming using servlets are to generate static content and Java Server Pages are used to generate dynamic content.
6. Creating a pure Dynamic Web Application which retrieves the data from Database according to the client request using JDBC.

Course Outcomes:

1. Create and Mange static web pages for given scenario

2. Apply server side technologies to establish dynamic applications

3. Implement web applications with effective data management

4. Develop secure web applications with session management API's

List of Experiments

1. Program to implement exceptions handling mechanism.
2. Program using applets.
3. Program to illustrate the use of RMI (Remote Method Invocation)
4. Basic Servlet program.
5. Program to demonstrate dynamic html using java Servlet.
6. Program to demonstrate cookies in java servlets.

7. Basic JSP program.
8. JSP program to implement verification of a particular user login and display a welcome page
9. Program to print even and odd numbers.
10. Program to verify the particular user and redirect to welcome page if credentials are valid else print proper message.
11. Session management in jsp.
12. Program for Custom JSP Tag.
13. Write a program to describe <jsp: param>, <jsp: include> and <jsp: forward> action.
14. Program for <jsp:useBean> Tag
15. Program for <jsp:plug-in> Tag
16. Database Operation in jsp
17. Program to update student Info using JDBC Connection
18. Program to insert data into database and retrieve data from database.

Text Books:

1. Web Technologies, Black Book, Kogent Learning Solutions Inc, Dreamtech Press.

2. JDBC, Servlets, and JSP, New Edition, Santhosh Kumar K, Kogent Learning Solutions Inc, Dreamtech Press.

Subject code	**Object Oriented Analysis and Design**	**L**	**T**	**P**	**Credits**
4010172401		**04**	**01**	**0**	**04**

Course Overview:

This course is intended to provide an in depth understanding of object oriented approaches to software development, in particular to the analysis and design phases of the software life cycle. Topics include notation, methods, competing methodologies, issues in object oriented development, and recent advancements which complement traditional object-oriented methodologies.

Course Objectives:

1. Essential and fundamental aspects of object-oriented analysis and design, in terms of "how to use" it for the purpose of specifying and developing software.
2. Explore and analyze different analysis and design models, such OO Models, Structured Analysis and Design Models, etc
3. Understanding the insight and knowledge into analyzing and designing software using different object-oriented modeling techniques
4. Understanding the fundamental principles through advanced concepts of analysis and design using UML

Course Outcomes:

1. Possess an ability to practically apply knowledge software engineering methods, such as object-oriented analysis and design methods with a clear emphasis on UML
2. Have a working ability and grasping attitude to design and conduct object-oriented analysis and design experiments using UML, as well as to analyze and evaluate their Models.
3. Have a capacity to analyze and design software systems, components to meet desired needs
4. Display an ability to identify, formulate and solve software development problems: software requirements, specification (problem space), Software design, and implementation (solution space).
5. Show an ability to use the graphical UML representation using tools, such as IBM's Rational Rose or Microsoft's Vision.

Unit-I:

Introduction to UML

The meaning of Object-Orientation, object identity, encapsulation, information hiding, polymorphism, generosity, importance of modeling, principles of modeling, object oriented modeling, conceptual model of the UML, Architecture.

Outcome:

1. Possess an ability to practically apply knowledge software engineering methods, such as object-oriented analysis and design methods with a clear emphasis on UML

Activity/Event :

Seminar on Case Study and Class test

Unit-II:

Basic structural Modeling & Class & object diagrams

Basic structural Modeling: Classes, relationships, common mechanisms, diagrams, Advanced structural modeling: advanced relationships, interfaces, types & roles, packages, instances.

Class & object diagrams: Terms, concepts, examples, modeling techniques, class & Object diagrams.

Outcome:

1. Have a working ability and grasping attitude to design and conduct object-oriented analysis and design experiments using UML, as well as to analyze and evaluate their Models.

Activity/Event :

Seminar on Case Study and Class test

Unit-III:

Sequence diagrams & Collaboration diagrams

Sequence diagrams: Terms, concepts, differences between collaboration and sequence diagrams, depicting synchronous messages with/without priority call back mechanism broadcast message.

Collaboration diagrams: Terms, Concepts, depicting a message, polymorphism in collaboration diagrams, iterated messages, use of self in messages.
Outcome:

1. Have a capacity to analyze and design software systems, components to meet desired needs

Activity/Event :

Seminar on Case Study and Class test

Unit-IV:

Behavioral Modeling & Advanced Behavioral Modeling

Behavioral Modeling:

Interactions, use cases, use case diagrams, activity diagrams.
Advanced Behavioral Modeling: Events and signals, state machines, processes & threads, time and space, state chart diagrams

Outcome:

1. Display an ability to identify, formulate and solve software development problems: software requirements, specification (problem space), Software design, and implementation (solution space).

Activity/Event :

Seminar on Case Study and Class test

Unit-V:

Architectural Modeling

Terms, concepts, examples, modeling techniques for component diagrams and deployment diagrams.

Outcome:

1. Show an ability to use the graphical UML representation using tools, such as IBM's Rational Rose or Microsoft's Vision

Activity/Event :

Seminar on Case Study and Class tests.

Text Books:

1. The Unified Modeling Language User Guide, Grady Booch, Rambaugh, Ivar Jacobson, PEA
2. Fundamentals of Object Oriented Design in UML, Meilir Page- Jones, Addison Wesley

Reference Books:

1. Head First Object Oriented Analysis &Design, Mclaughlin, SPD OReilly,2006
2. Object oriented Analysis& Design Using UML, Mahesh ,PHI
3. The Unified Modeling Language Reference Manual, 2/e, Rambaugh, GradyBooch,etc., PEA
4. Object Oriented Analysis & Design, Satzinger, Jackson, Thomson
5. Object Oriented Analysis Design & implementation, Dathan.,Ramnath, University Press
6. Object Oriented Analysis & Design, John Deacon, PEA
7. Fundamentals of Object Oriented Analysis and Design in UML, M Pages-Jones, PEA
8. Object-Oriented Design with UML, Barclay, Savage, Elsevier,2008

Subject Code	Computer Networks	L	T	P	Credits
4010172402		04	01	0	04

Course Overview:

This course introduces basic routing algorithms and related protocols for Network Management and define the functionality of different type of network layers and it efficient usage.

Course Objectives:

Students will learn the basics of networks and its topological design.
Students will learn different type of error detection and control for data link layer protocol and understanding the different type of routing algorithm.

Course Outcomes:

1. Student will able to Solve error detection and correction techniques of data link layer and describe MAC layer techniques and to get a concept of different type of routing algorithm and channel allocation method for communication
2. student will able to full fill the network security needs and its implementation in network management system and demonstration of WWW and multimedia

Unit-I:

Network reference model

Introduction to Networks, Network H/w, OSI-reference models, Network Standardization, Transmission media, Narrowband ISDN, Broad band ISDN, ATM. **Physical Layer:** Guided transmission media – Magnetic media, Twisted Pair, coaxial cable, fiber optics.

Outcome:

Student will understand the seven layers of interface and its functionality with different communication mode.

Activity/Event :

You tube tutorial

Unit-II:

The data Link layer

Design Issues, Error detection and correction, Elementary Data Link Protocols, Sliding window protocols in HDLC.

Channel allocation methods: TDM, FDM, ALOHA, Carrier sense Multiple access protocols, Collision Free protocols – IEEE standard BO2 for LANS – Ethernet, Token Bus, Token ring, Bridges.

Network layer Routing Algorithms: Shortest path, Flooding, Flow based Distance vector, Link state, Hierarchical, DSR, AODV, Broadcast routing, Congestion Control algorithms-General principles of congestion control, Congestion prevention polices, Choke packets and Load shedding.

Outcome:

Student will able to know the different type of error and flow control with different technique

Activity/Event :

Seminar

Unit-III:

Internet Working

Network layer Design issue Tunneling, internet working, Fragmentation, network layer in the internet – IP protocol s, IP address, Subnets, ICMP, BOP, internet multicasting ,Mobile IP . Network layer in the ATM Networks –cell formats , connection setup, routing and switching , service categories and quality of service ,ATM LAN s.

Outcome:

Student will understand the Routing algorithms and their addressing modes.

Activity/Event :

Seminar followed by test

Unit-IV:

The Transport Layer

Elements of transport protocols – addressing, establishing a connection, releasing connection, flow control and buffering and crash recovery, end to end protocols: UDP, reliable Byte Stream (TCP) end to end format, segment format, connection establishment and termination, sliding window revisited, adaptive retransmission, TCP extension, Remote Procedure Call Performance Issues.

Outcome:

Student will understand the different type of protocols such as TCP, UDP, BGP, ICMP, POP etc.

Activity/Event :

Seminar followed by test.

Unit-V:

Application Layer:

Network Security. Security Mechanisms: Authentication Protocols, Firewalls, Cryptographic

Algorithms: DES, RSA, Digital Signatures, Domain Naming system Hierarchy, Name servers. Traditional Applications: SMTP, MIME, World Wide Web: HTTP, Management of public keys.

Outcome :

Student will able to understand the functionality interface with user along with protocol

Activity/Event :

Tutorial

Text Books:

Computer Networks Andrew, Tanenbaum, 4/e, Pearson

Reference Books:

Data communications and networking Forouzan, 4/e, TMH

Subject Code	Data Warehousing and Data Mining	L	T	P	Credits
4010172403		04	01	0	04

Course Overview:

This course introduces basic Data ware housing structure and its structural design and functionality of different type of Data mining and its efficient usage.

Course Objectives:

Students will learn the basics of Data ware housing and its application.

Course Outcomes:

1. Student will able to know various types of data and how to maintain the quality of data.
2. Students will have the knowledge of different structure of Web data, search engine and web mining

Unit-I:

Introduction to System Introduction to Data mining

Introduction to Data mining, types of Data, Data Quality, Data Processing, Measures of Similarity and Dissimilarity, Exploring Data: Data Set, Summary Statistics, Visualization, OLAP and multi dimensional data analysis.

Outcome:

Student will able to define the basic structure of data mining and its applications.

Activity/Event :

Seminar by toppers of the class

Unit-II:

Classification

Basic Concepts, Decision Trees and model evaluation: General approach for solving a classification problem, Decision Tree induction, Model over fitting: due to presence of noise, due to lack of representation samples, Evaluating the performance of classifier. Nearest Neighborhood classifier, Bayesian Classifier, Support vector Machines: Linear SVM, Separable and Non Separable case.

Outcome:

Student will able to define the features for classification and prediction along with framing the decision tree.

Activity/Event :

Group seminar for deep discussion

Unit-III:

Association Analysis:

Problem Definition, Frequent Item-set generation, rule generation, compact representation of frequent item sets, FP-Growth Algorithms. Handling Categorical, Continuous attributes, Concept hierarchy, Sequential, Sub graph patterns.

Outcome:

Student will able to define the application of association and support and confidence analysis for trade analysis.

Activity/Event :

Selected members of the class will give seminar on the same topic followed by test.

Unit-IV:

Clustering

Over view, K-means, Agglomerative Hierarchical clustering, DBSCAN, Cluster evaluation: overview, Unsupervised Cluster Evaluation using cohesion and separation, using proximity matrix, Scalable Clustering algorithm.

Outcome:

Student will able to define the necessity of cluster and group formation in real time

Activity/Event :

Seminar and followed by test

Unit-V:

Web data mining

Introduction, Web terminology and characteristics, Web content mining, Web usage mining, web structure mining, Search Engines: Characteristics, Functionality, Architecture, Ranking of WebPages, Enterprise search

Outcome:

Student will able to understand the basic structure of internet and how to do mining and search engine functionality.

Activity/Event :

Seminar & test

Text Books:

1. Introduction to Data Mining: Pang-Ning tan, Michael Steinbach, Vipin kumar, Addision- Wesley.

Reference Books:

1. Data Mining: Introductory and Advanced Topics, Margaret H Dunham, Pearson, 2008.

Subject Code	Mobile Computing (Elective-1)	L	T	P	Credits
4010172404		04	01	0	04

Course Overview:

This course introduces basic Mobile Network structure and it working topological design. Routing algorithms and related protocols for MANETS

Course Objectives:

Data Mining: Introductory and Advanced Topics, Margaret H Dunham, Pearson, 2008.

Course Outcomes:

Student will able to know various Wireless Network Models and classify the functionality of every individual layer. Classifying the different type of Adhoc topology and transmission of controlling the data on every layer

Unit-I:

Mobile Communications

An Overview- Mobile Communication-guided transmission, unguided transmission- signal propagation frequencies, antennae, modulation, modulation methods and standards for voice-oriented data communication standards, modulation methods and standards for data and voice communication, mobile computing- novel applications and limitations, mobile computing architecture, mobile system networks. Mobile devices and systems: Cellular networks and frequency reuse, Mobile smart phones, Smart mobiles and systems, handheld pocket computers, Handheld devices, Smart systems, Limitations of mobile devices

Outcome:

To know about the cellular technology with GSM

Activity/Event :

To show the real time model with you tube and ppt

Unit-II:

GSM and other 2G Architectures:

GSM and other 2G Architectures: GSM-services and system architecture, Radio interfaces of GSM, Protocols of GSM, Localization, Call handling, GPRS system architecture. Wireless medium access control, CDMA, 3G,and4G Communication: Modulation, Multiplexing, Controlling the medium access, Spread spectrum, Coding methods, IMT-20003G wireless communication standards, WCDMA 3G communication standards, CDMA 3G communication standards, Broadband wireless access, 4Gnetworks.

Outcome:

To know about the GSM Technology and cellular architecture

Activity/Event:

Discuss about the cellular architecture formation with you tube video

Unit-III:

Mobile IP Network layer:

Mobile IP Network layer: IP and Mobile IP network layers: OSI layer functions, TCP/IP and Internet protocol, Mobile internet protocol; Packet delivery and Handover Management; Location Management: Agent Discovery; Mobile TCP Introduction to Mobile Adhoc network: fixed infrastructure architecture, MANET infrastructure architecture; MANET: properties, spectrum, applications; Security in Ad-hoc network; Wireless sensor networks; sensor network applications.

Outcome:

Student will able to define the Internet Protocol and its implementation

Activity/Event :

You tube demonstration

Unit-IV:

Synchronization

Synchronization: Synchronization in mobile computing systems, Usage models for Synchronization in mobile application, Domain-dependant specific rules for data synchronization, Personal information manager, synchronization and conflict resolution strategies, synchronizer; Mobile agent: mobile agent design, aglets; Application Server

Outcome:

To know about the synchronization and clock signaling with cells.

Activity/Event :

Showing the demo of cells handover and handshaking synchronization with you tube .

Unit-V:

Mobile Wireless

Mobile Wireless Short Range Networks and Mobile Internet: Wireless networking and wireless LAN, Wireless LAN (WLAN) architecture, IEEE 802.11protocol layers, Wireless application protocol (WAP)-WAP1.1 architecture, wireless datagram protocol (WDP), Wireless Transport Layer Security (WTLS), wireless transaction and session layers, wireless application environment.

Outcome:

Student will learn the wireless architecture and layers

Activity/Event :

YouTube Demonstration for the wireless and sensor technology

Text Books:

1. Raj Kamal, “Mobile Computing,” second edition, Oxford.
2. Technology Applications and Service Creation” Second Edition, Mc Graw Hill.

Reference Books:

1. Asoke k talukder, hasanahmed, roopa r yavagal,“mobile computing,

Subject Code	Human Computer Interaction (Elective-1)	L	T	P	Credits
4010172405		04	01	0	04

Course Overview:

This course covers the principles of human-computer interaction and the design and evaluation of user interfaces. Topics include an overview of human information processing subsystems (perception, memory, attention, and problem solving); how the properties of these systems affect the design of user interfaces; the principles, guidelines, and specification languages for designing good user interfaces.

Course Objectives:

1. Demonstrate an understanding of guidelines, principles, and theories influencing human computer interaction.
2. Recognize how a computer system may be modified to include human diversity
3. Design mock ups and carry out user and expert evaluation of interface
4. Design mock ups and carry out user and expert evaluation of interfaces
5. Carry out the steps of experimental design, usability and experimental testing, and evaluation of human computer interaction systems
6. Use the information sources available, and be aware of the methodologies and technologies supporting advances in HCI.

Course Outcomes:

1. Apply the basics of human and computational abilities and limitations
2. Apply new theories, tools and techniques in HCI.
3. Have a capacity to analyze and design software systems, components to meet desired needs.
4. Apply the fundamental aspects of designing and evaluating interfaces.
5. Practice a variety of simple methods for evaluating the quality of a user interface
6. Apply appropriate HCI techniques to design systems that are usable by people.

Unit-I:

Introduction to HCI & The graphical user interface

Introduction:

Importance of user Interface, definition, importance of good design. Benefits of good design. A brief history of Screen design.

The graphical user interface:

Popularity of graphics, the concept of direct manipulation, graphical system, Characteristics, Web user –interface popularity, characteristics- Principles of user interface.

Outcome:

1. Understand the basics of human and computational abilities and limitations.
2. Understand basic theories, tools and techniques in HCI.

Activity/Event on Unit-1: Seminar and Class Test

Unit-II:

Design process

Design process: Human interaction with computers, importance of human characteristics human consideration, Human interaction speeds, Understanding business junctions.

Outcome:

1. Have a capacity to analyze and design software systems, components to meet desired needs.

Activity/Event:

Seminar and Class Test

Unit-III:

Screen Designing :

Design goals, Screen planning and purpose, organizing screen elements, ordering of screen data and content, screen navigation and flow, Visually pleasing composition, amount of information, focus and emphasis, presentation information simply and meaningfully, information retrieval on web, statistical graphics, Technological consideration in interface design.

Outcome:

1. Understand the fundamental aspects of designing and evaluating interfaces.

Activity/Event :

Seminar and Class Test

Unit-IV:

Windows and Components

Windows: Windows new and Navigation schemes selection of window, selection of devices based and screen based controls.

Components: Components text and messages, Icons and increases, Multimedia, colors, uses problems, choosing colors.

Outcome:

1. Practice a variety of simple methods for evaluating the quality of a user interface

Activity/Event:

Seminar and Class Test

Unit-V:

Software tools and Interaction Devices

Software tools :
Specification methods, interface, Building Tools.

Interaction Devices: Keyboard and function keys, pointing devices, speech recognition digitization and generation, image and video displays, drivers..

Outcome:

1. Apply appropriate HCI techniques to design systems that are usable by people

Activity/Event :

Seminar and Class Test

Text Books:

1. Human Computer Interaction. 3/e, Alan Dix, Janet Finlay, Goryd, Abowd, Russell Beal, PEA,2004
2. The Essential guide to user interface design,2/e, Wilbert O Galitz, Wiley Dreama Tech

Reference Books:

1. Designing the user interface. 4/e, Ben Shneidermann, PEA
2. User Interface Design, Soren Lauesen , PEA.
3. Interaction Design PRECE, ROGERS, SHARPS, Wiley
4. Human Computer, Interaction Dan R.Olsan, Cengage ,2010

Subject Code	Cloud Computing (Elective-1)	L	T	P	Credits
4010172406		04	01	0	04

Course Overview:

This course introduces basic functionality of different type of services provided by the cloud And cloud structure and it working topological design.

Course Objectives:

1. Students will learn the basics clouds and its applications,
2. Students will learn the key concept IAAS, PAAS, SAAS, along with security issues related to cloud data

Course Outcomes:

Student will able to
Evaluate the different view of virtual servers and its computing collaboration with different applications with virtual work environment..

Unit-I:

Introduction to Cloud & Parallel and Distributed Systems

Network centric computing, Network centric content, peer-to –peer systems, cloud computing delivery models and services, Ethical issues, Vulnerabilities, Major challenges for cloud computing

Parallel and Distributed Systems:

Introduction, architecture, distributed systems, communication protocols, logical clocks, message delivery rules, concurrency, model concurrency with Petri Nets

Outcome:

Student will able to understand the basic architecture of Distributed system

Activity/Event :

Sseminar

Unit-II:

Cloud Infrastructure:

At Amazon, The Google Perspective, Microsoft Windows Azure, Open Source Software Platforms, Cloud storage diversity, Inter cloud, energy use and ecological impact, responsibility sharing, user experience, Software licensing

Cloud Computing : Applications and Paradigms: Challenges for cloud, existing cloud applications and new opportunities, architectural styles, workflows, The Zookeeper, The Map Reduce Program model, HPC on cloud, biological research

Outcome:

Student will able to understand the parallel computing and its functionality

Activity/Event :

Seminar followed by test.

Unit-III:

Cloud Resource virtualization:

Cloud Resource Management and Scheduling
Virtualization, layering and virtualization, virtual machine monitors, virtual machines, virtualization- full and para, performance and security isolation, hardware support for virtualization, Case Study: Xen, vBlades

Cloud Resource Management and Scheduling: Policies and Mechanisms, Applications of control theory to task scheduling, Stability of a two-level resource allocation architecture, feed back control based on dynamic thresholds, coordination, resource bundling, scheduling algorithms, fair queuing, start time fair queuing, cloud scheduling subject to deadlines, Scheduling Map Reduce applications, Resource management and dynamic application scaling

Outcome:

Student will able to know how the resource stored on the cloud and its usage

Activity/Event :

Seminar by toppers

Unit-IV:

Storage Systems & Cloud Security:

Storage Systems: Evolution of storage technology, storage models, file systems and database, distributed file systems, general parallel file systems. Google file system. Apache Hadoop, Big Table, Megastore (text book 1), Amazon Simple Storage Service(S3) (Text book 2)

Cloud Security: Cloud security risks, security – atop concern for cloud users, privacy and privacy impact assessment, trust, OS security, Virtual machine security, Security risks

Outcome:

Student will able to know the different security mechanism to secure the data.

Activity/Event :

Seminar on Cryptography

Unit-V:

Cloud Application Development (Google Microsoft):

Storage Systems:

Evolution of storage technology, storage models, file systems and database, distributed file systems, general parallel file systems. Google file system. Apache Hadoop, Big Table, Megastore (text book 1), Amazon Simple Storage Service(S3) (Text book 2)

Cloud Security:

Cloud security risks, security – atop concern for cloud users, privacy and privacy impact assessment, trust, OS security, Virtual machine security, Security risks

Outcome:

Student will understand the Cloud server development and allocation to its users

Activity/Event :

Tutorial

Text Books:

1. Cloud Computing, Theory and Practice, Dan C Marinescu, MK Elsevier

Reference Books:

1. Mastering Cloud Computing, Foundations and Application Programming, Raj Kumar, Buyya, Christen vecctiola, S Tammarai selvi, TMH

Subject Code	SOFTWARE PROJECT MANAGEMENT (Elective-2)	L	T	P	Credits
4010172407		4	1	0	4

Course Overview:

Students will study the software project planning, cost estimation and scheduling, project management tools, factors influencing productivity and success. Students will also learn productivity metrics, analysis of options and risks, software process standards and process implementation, software contracts and intellectual property and approaches to maintenance and long term software development

Course Objectives:

1. Understand the fundamental principles of Software Project management & will also have a good knowledge of responsibilities of project manager
2. You will have good knowledge of the issues and challenges faced while doing the Software project Management and be familiar with the different methods and techniques used for project management
3. Will be able to do the Project Scheduling, tracking, Risk analysis, Quality management and Project Cost estimation using different techniques

Course Outcomes:

1. Basic Knowledge on software Engineering
2. Basic knowledge on project goals
3. Basic Knowledge on Software Constraints

Unit-I:

Conventional Software Management:

The waterfall model, conventional software
Management performance.

Evolution of Software Economics:
Software Economics, pragmatic software cost estimation.

Improving Software Economics:
Reducing Software product size, improving software
processes, improving team effectiveness, improving automation, Achieving required quality, peer inspections.

The old way and the new: The principles of conventional software Engineering, principles of modern software management, transitioning to an iterative process.

Outcomes:

1. Understand the importance of SPM
2. Differentiate software projects and other types of project

Activity/Event :

Seminar

Unit-II:

Life cycle phases: Engineering and production stages, inception, Elaboration, construction, transition phases.

Artifacts of the process: The artifact sets, Management artifacts, Engineering artifacts, programmatic artifacts

Outcome:

1. Align the project to the organization's strategic plans and business justification throughout its lifecycle

Activity/Event :

Seminar

Unit-III:

Model based software architectures:

A Management perspective and technical perspective.

Work Flows of the process:

Software process workflows, Iteration workflows.

Checkpoints of the process: Major

Mile stones, Minor Milestones, Periodic status assessments.

Iterative Process Planning:

Work breakdown structures, planning guidelines, cost and schedule estimating, Iteration planning process, Pragmatic planning.

Outcomes:

1.The Student Estimates projects techniques using a bottom-up approach

2.The student Counts the function point for a system

Activity/Event :

Case study

Unit-IV:

Project Organizations and Responsibilities:

Line-of-Business Organizations, Project Organizations, evolution of Organizations.

Process Automation:
Automation Building blocks, The Project Environment.

Project Control and Process instrumentation:
The seven core Metrics, Management indicators, quality indicators, life cycle expectations, pragmatic Software Metrics, Metrics automation.

Outcomes:

1.Adapt projects in response to issues that arise internally and externally.

2.Interact with team and stakeholders in a professional manner, respecting differences, to ensure a collaborative project environment

Activity/Event :

Case study

Unit-V:

Tailoring the Process:

Process discriminates.

Future Software Project Management:

Modern Project Profiles, Next generation Software economics, modern process transitions.

Outcome:

1. Implement project management knowledge, processes, lifecycle and the embodied concepts, tools and techniques in order to achieve project success

Activity/Event :

Seminar

Text Books:

Software Project Management, Walker Royce, PEA, 2005

Reference Books: .

1.Software Project Management, Bob Hughes,3/e, Mike Cotterell, TMH

2. Software Project Management, Joel Henry, PEA

3. Software Project Management in practice, Pankaj Jalote, PEA, 2005

Subject Code	ARITIFICIAL INTELLIGENCE (ELECTIVE-2)	L	T	P	Credits
4010172408		04	01	0	04

Course Overview:

This is an introductory course on Artificial Intelligence. The topics may include: AI methodology and fundamentals; intelligent agents; search algorithms; game playing; supervised and unsupervised learning; decision tree learning; neural networks; nearest neighbour methods; dimensionality reduction; clustering; kernel machines; support vector machines; uncertainty and probability theory; probabilistic reasoning in AI; Bayesian networks; statistical learning; fuzzy logic.

Course Objectives:

1. To have a basic proficiency in a traditional AI language including an ability to write simple to intermediate programs and an ability to understand code written in that language.
2. To have an understanding of the basic issues of knowledge representation and blind and heuristic search, as well as an understanding of other such as minimax, resolution etc. that play an important role in AI programs
3. To have a basic understanding of some of the more advanced topics of AI such as learning, natural language processing, agents and robotics, expert systems, and planning.

Course Outcomes:

1. Identify problems that are amenable to solution of AI methods, and which AI methods may be suited to solving a given problem.
2. Formalize a given problem in the language/framework of different AI methods
3. Implement basic AI algorithms
4. Design and carry out an empirical evaluation of different algorithms on a problem formalization, and state the conclusions that the evaluation supports.

Unit-I:

Introduction to artificial intelligence

Introduction , history, intelligent systems, foundations of AI, applications, tic-tac-tie game playing, development of ai languages, current trends in AI.

Problem solving: state-space search and control strategies: Introduction, general problem solving, characteristics of problem, exhaustive searches, heuristic search techniques, iterative- deepening a*, constraint satisfaction

Outcome:

1. Identify problems that are amenable to solution of AI methods, and which AI methods may be suited to solving a given problem.

Activity/Event :

Seminar and Class Test

Unit-II:

Problem reduction and game playing

Introduction, problem reduction, game playing, alpha- beta pruning, two-player perfect information games **Logic concepts:** Introduction, propositional calculus, proportional logic, natural deduction. system, axiomatic system, semantic tableau system in proportional logic, resolution refutation in proportional logic, predicate logic

Outcome:

1. Formalize a given problem in the language/framework of different AI methods

Activity/Event :

Seminar and Class Test

Unit-III:

Knowledge representation:

Introduction, approaches to knowledge representation, knowledge representation using semantic network, extended semantic networks for KR, knowledge representation using frames.

advanced knowledge representation techniques: Introduction, conceptual dependency theory, script structure, cyc theory, case grammars, semantic web.

Expert system and applications:

Introduction phases in building expert systems, expert system versus traditional systems, rule-based expert systems blackboard systems truth maintenance systems, application of expert systems, list of shells and tools

Outcome:

1. Student can develop rule based applications

Activity/Event :

Seminar and Class Test

Unit-IV:

Uncertainty measure: probability theory:

Introduction, probability theory, Bayesian belief networks, certainty factor theory, dempster-shafer theory

Fuzzy sets and fuzzy logic: Introduction, fuzzy sets, fuzzy set operations, types of membership functions, multi valued logic, fuzzy logic, linguistic variables and hedges, fuzzy propositions, inference rules for fuzzy propositions, fuzzy systems.

Outcome:

1. Student will understand the fuzzy logics

Activity/Event :

Seminar and Class Test

Unit-V:

Machine learning paradigms:

Introduction, machine learning systems, supervised and unsupervised learnings, inductive learning, deductive learning, clustering, support vector machines, case based reasoning and learning

Artificial neural networks:

Introduction, artificial networks, single layer feed forward networks, multi layered forward networks, design issues of artificial neural networks

Outcome:

Student can implement machine learning concepts

Activity/Event :

Seminar and Class Test

Text Books:

1. Artificial Intelligence- Saroj Kaushik, CENGAGE Learning
2. Artificial intelligence, A modern Approach , 2nd ed, Stuart Russel, Peter Norvig, PEA
3. Artificial Intelligence- Rich, Kevin Knight,Shiv Shankar B Nair, 3rd ed, TMH
4. Introduction to Artificial Intelligence, Patterson, PHI

Reference Books:

1. Artificial intelligence, structures and Strategies for Complex problem solving, -George F Lugar, 5th ed, PEA
2. Artificial Intelligence, A new Synthesis, Nils J Nilsson, Elsevier
3. Introduction to Artificial Intelligence, Ertel, Wolf Gang, Springer

Subject Code	EMBEDDED SYSTEM (Elective-2)	L	T	P	Credits
4010172409		04	01	0	04

Course Overview:

This course introduces basic designing electronic circuits for the processing of information in communications and control systems and The ability to analyze, design, test and maintain complex embedded systems.

Course Objectives:

Discuss the major components that constitute an embedded system. and Implement small programs to solve well-defined problems on an embedded platform

Course Outcomes:

1. Exploration and analysis of various types of timers and Universal Asynchronous Receiver/ Transmitter.
2. Student will able to do deep discussion about message queues, mailboxes, and pipes. Describe the process of effective memory management

Unit-I:

Introduction to Embedded systems:

What is an embedded system Vs. General computing system, history, classification, major application areas, and purpose of embedded systems. Core of embedded system, memory, sensors and actuators, communication interface, embedded firmware, other system components, PCB and passive components.

Outcome:

Student will able to know the basic structure of Embedded system

Activity/Event :

Seminar

Unit-II:

Microcontroller

8—bit microcontrollers architecture:

Characteristics, quality attributes application specific, domain specific, embedded systems. Factors to be considered in selecting a controller, 8051 architecture, memory organization, registers, oscillator unit, ports, source current, sinking current, design examples.

Outcome:

Student will able to know different type of Controller and its structure

Activity/Event :

Seminar and test

Unit-III:

Scheduling

RTOS and Scheduling, Operating basics, types, RTOS, tasks, process and threads, multiprocessing and multitasking, types of multitasking, non preemptive, preemptive scheduling.

Outcome:

Student will be able know to different type of Synchronous and Asynchronous with algorithm

Activity/Event :

Seminar

Unit-IV:

Communication and Deadlocks

Task communication of RTOS, Shared memory, pipes, memory mapped objects, message passing, message queue, mailbox, signaling, RPC and sockets, task communication/synchronization issues, racing, deadlock, live lock, the dining philosopher's problem.

Outcome:

Student will able to know characteristics , detection and recovery from deadlock.

Activity/Event :

Tutorial

Unit-V:

Synchronization

The producer-consumer problem, Reader writers problem, Priority Inversion, Priority ceiling, Task Synchronization techniques, busy waiting, sleep and wakery, semaphore, mutex, critical section objects, events, device, device drivers, how to clause an RTOS, Integration and testing of embedded hardware and fire ware. Simulators, emulators, Debuggers, Embedded Product Development life cycle (EDLC), Trends in embedded Industry, Introduction to ARM family of processor.

Outcome:

Student will able to understand and implementation of the DMA, Asy & Syn mode

Activity/Event :

Tutorial

Text Books:

1. Introduction to embedded systems Shibu. K.V, TMH, 2009.

Reference Books:

2. Embedded Systems, Rajkamal, TMH, 2009.
3. Embedded Software Primer, David Simon, Pearson

Subject Code		L	T	P	Credits
4010172421	**SOFT SKILLS LAB**	**0**	**0**	**3**	**2**

Course Overview:

To impart soft skills like positive attitude, self management, body language, assertiveness, goal setting, team work, adaptability, integrity, leadership for students to become the best professional with ethical values.

Course Objectives:

To inculcate work ethics and to develop interpersonal relations through emotional intelligence to the learners and make them enhance employability **skills and work life balance**

Course Outcomes:

- The learner will be able to maintain work life balance and will become professionally and ethically sound in overcoming stress.
- The learner will be able to maintain interpersonal relationships by managing emotional intelligence.
- The learner will be able to acquire employability and problem solving skills.

Experiments:

Body language, 2. Life skills 3. Emotional intelligence 4. Problem solving skills

5.Employability skills

1.Body Language

a. Facial Expressions.
b. Kinesics.
c. Oculesics.
d. Haptics.
e. Proxemics
f. Para Linguistics.

2.Life Skills

a. Positive Attitude
b. Social Behaviour &Social Norms.
c. Ethics, Values and Positive Work Ethics.
d. Time Management
e. Goal Setting: Short term, Long Term.

3.Emotional Intelligence

g. Self Awareness through Johari Window and SWOT analysis.
h. Self Control.
i. Self Motivation.
j. Empathy.
k. Social Skills.
l. Self Esteem.
m. Managing stress.
n. Assertiveness.

4.Problem Solving Skills

o. Critical Thinking and Brain Storming
p. Lateral Thinking and Six Thinking Hats.
q. Creative Thinking.
r. Conflict Management.

5.Employability Skills

s. Group Discussion.
t. Team Building and Leadership Qualities
u. Interview Skills.

Reference Books:

1. “The Definitive Book Of Body Language”, Allan & Barbara Pease
2. “You Can Win”, Shiv Khera.
3. ““Lateral Thinking”, Edward De Bono.
4. “How To Prepare For Group Discussions And Interview”, Hari Mohan Prasad, Rajnish Mohan, 2nd Edition, TMH.
5. “Emotional Intelligence”, Daniel Goleman.
6. “ The 7 Habits Of Highly Effective People“, Stephen R. Covey
7. “Working in Teams”, Sandy Pokras.

Subject Code	DATA WAREHOUSING AND DATA MINING LAB	L	T	P	Credits
4010172422		0	0	3	2

Course Overview:

The main objective of this lab is to impart the knowledge on how to implement classical models and algorithms in data warehousing and data mining and to characterize the kinds of patterns that can be discovered by association rule mining, classification and clustering. At the end to compare and contrast different conceptions of data mining.

Course Objectives:

1. Students will learn the basics of Data ware housing and its application.
2. Students will learn different type of Data mining Algorithms.
3. The main objective of this lab is to impart the knowledge on how to implement classical models and algorithms in data warehousing and data mining.
4. To characterize the kinds of patterns that can be discovered by association rule mining, classification and clustering. At the end to compare and contrast different conceptions of data mining.

Course Outcomes:

- To understand the basic principles, concepts and applications of data warehousing and data mining,
- Ability to do Conceptual, Logical, and Physical design of Data Warehouses OLAP applications and OLAP deployment.
- Have a good knowledge of the fundamental concepts that provide the foundation of data mining.

Detailed Syllabus :

1. Demonstration of preprocessing on dataset student. arff
2. Demonstration of preprocessing on dataset labor. arff
3. Demonstration of Association rule process on dataset contact lenses. arff using apriority algorithm
4. Demonstration of Association rule process on dataset test. arff using apriority algorithm
5. Demonstration of classification rule process on dataset student. arff using j48 algorithm
6. Demonstration of classification rule process on dataset employee .arff using j48 algorithm
7. Demonstration of classification rule process on dataset employee. arff using id3 algorithm
8. Demonstration of classification rule process on dataset employee. arff using naïve bayes algorithm
9. Demonstration of clustering rule process on dataset iris. arff using simple k-means
10. Demonstration of clustering rule process on dataset student. arff using simple k-means

Reference Books:

1. Ramesh Sharda, Dursun Delen, David King Business Intelligence, 2/E; Efraim Publisher Turban,pearson Education, 2011 2.
2. Berry, Gordon S. Linoff, "Data Mining Techniques: For Marketing, Sales, and Customer Relationship Management", John Wiley & Sons Inc publishers, 3 rd Edition, 2011.

Subject Code	Object Oriented Analysis and Design Lab	L	T	P	Credits
4010172423		0	0	3	2

Course Overview:

The analysis, design, coding, documentation, database design of mini project which will be carried out in 2nd year can be done in object-oriented approach using UML and by using appropriate software which supports UML.

Course Objectives:

- Construct UML diagrams for static view and dynamic view of the system.
- Generate creational patterns by applicable patterns for given context.
- Create refined model for given Scenario using structural patterns.
- Construct behavioral patterns for given applications.

Course Outcomes:

- Understand the Case studies and design the Model.
- Understand how design patterns solve design problems.
- Develop design solutions using creational patterns.
- Construct design solutions by using structural and behavioral patterns.

Detailed Syllabus :

Week 1:

Familiarization with Rational Rose or Umbrella
For each case study:

Week 2, 3 & 4:

For each case study:

a) Identify and analyze events
b) Identify Use cases
c) Develop event table
d) Identify & analyze domain classes
e) Represent use cases and a domain class diagram using Rational Rose
f) Develop CRUD matrix to represent relationships between use cases and problem domain classes

Week 5 & 6:

For each case study:
a) Develop Use case diagrams
b) Develop elaborate Use case descriptions & scenarios
c) Develop prototypes (without functionality)
d) Develop system sequence diagrams

Week 7, 8, 9 & 10:

For each case study:
a) Develop high-level sequence diagrams for each use case
b) Identify MVC classes / objects for each use case
c) Develop Detailed Sequence Diagrams / Communication diagrams for each use case showing interactions among all the three-layer objects
d) Develop detailed design class model (use GRASP patterns for responsibility assignment)
e) Develop three-layer package diagrams for each case study

Week 11 & 12:

For each case study:

a) Develop Use case Packages

b) Develop component diagrams

c) Identify relationships between use cases and represent them

d) Refine domain class model by showing all the associations among classes

Week 13 onwards:

For each case study:

a) Develop sample diagrams for other UML diagrams - state chart diagrams, activity diagrams and
deployment diagrams

Text Books:

1. Craig Larman, “Applying UML and Patterns: An Introduction to ObjectOriented Analysis and Design and the Unified Process”, 2nd ed., Pearson Education Asia, 2002.

Reference Books:

1. Simon Sennet, Steve McRobb, and Ray Farmer, “Object Oriented Systems Analysis and Design using UML”, 2nd ed., McGraw Hill, 2002.
2. Andrew Haigh, “Object-Oriented Analysis & Design,” 1st ed., Tata McGraw-Hill, 2001.

Subject Code	Mini Project	L	T	P	Credits
4010172431		0	0	0	2

For the Academic Year 2019-20

(V Semester)

Subject Code	BIG DATA ANALYTICS	L	T	P	Credits
4010173501		4	0	0	3

Course Overview:

Students should have knowledge of

- Learn Big Data Analytics
- Store, manage, and analyze unstructured data
- Query large data sets in near real time with Pig and Hive

Course Objectives:

- Introduce students the concept and challenge of big data (3 V's: volume, velocity, and variety).
- Teach students in applying skills and tools to manage and analyze the big data.

Course Outcomes:

After completion of course students will be able to

- Understand the concepts of Big data and challenges in processing Big Data
- Understand Hadoop architecture and eco-system.
- Gainconceptual understanding of Hadoop Distributed File System.
- Understand the concepts of map and reduce and functional programming
- Identify appropriate techniques and tools to solve actual Big Data problems.

CO-PO Mapping:

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1				2				1	1			
CO2	3			3				1	1			
CO3								2	1			
CO4	1								1	2		
CO5	2	2	2	2				2		2		

Unit-I:

Data structures in Java

Linked List, Stacks, Queues, Sets, Maps; Generics: Generic classes and Type parameters, Implementing Generic Types, Generic Methods, Wrapper Classes, Concept of Serialization.

Outcome:

- Understand basic data structures such as arrays, linked lists, stacks and queues.
- Define and working with Generic & Wrapper classes

Activity/Event:

Programming Contest

Unit-II:

Working with Big Data

Google File System, Hadoop Distributed File System (HDFS) – Building blocks of Hadoop (Name node, Data node, Secondary Name node, Job Tracker, Task Tracker), Introducing and Configuring Hadoop cluster (Local, Pseudo-distributed mode, Fully Distributed mode), Configuring XML files.

Outcome:

- Define and compare/contrast Google file system and HDFS.
- Understand how to configuring Hadoop cluster

Activity/Event:

Test / Seminar

Unit-III:

Writing Map Reduce Programs :

A Weather Dataset, Understanding Hadoop API for Map Reduce Framework (Old and New), Basic programs of Hadoop Map Reduce: Driver code, Mapper code, Reducer code, Record Reader, Combiner, Partitioner

Outcome:

- Explain about Map reducing

Activity/Event :

Practicing real time application.

Unit-IV:

Hadoop I/O :

The Writable Interface, Writable Comparable and comparators, Writable Classes: Writable wrappers for Java primitives, Text, Bytes Writable, Null Writable, Object Writable and Generic Writable, Writable collections, Implementing a Custom Writable: Implementing a Raw Comparator for speed, Custom comparators.

Outcome:

- Discuss on writable classes and comparators.

Activity/Event :

Programming Contest

Unit-V:

Pig

Hadoop Programming Made Easier Admiring the Pig Architecture, Going with the Pig Latin Application Flow, Working through the ABCs of Pig Latin, Evaluating Local and Distributed Modes of Running Pig Scripts, Checking out the Pig Script Interfaces, Scripting with Pig Latin Applying Structure to Hadoop Data with Hive: Saying Hello to Hive, Seeing How the Hive is Put Together, Getting Started with Apache Hive, Examining the Hive Clients, Working with Hive Data Types, Creating and Managing Databases and Tables, Seeing How the Hive Data Manipulation Language Works, Querying and Analyzing Data.

Unit-V :

- Usage of Hadoop programming
- Explain basics of Pig Latin Application
- Learned Pig Architecture

Activity/Event:

Test and Seminar

Text Books:

1. Big Java 4th Edition, Cay Horstmann, Wiley John Wiley & Sons, INC.
2. Hadoop: The Definitive Guide by Tom White, 3rd Edition, O'reilly.
3. Hadoop in Action by Chuck Lam, MANNING Publ.
4. Hadoop for Dummies by Dirk deRoos, Paul C.Zikopoulos, Roman B.Melnyk,Bruce Brown, Rafael Coss

Reference Books:

1. Hadoop in Practice by Alex Holmes, MANNING Publ.
2. Hadoop MapReduce Cookbook, Srinath Perera, Thilina Gunarathne.

Subject Code	**Network Programming**	**L**	**T**	**P**	**Credits**
4010173502		**04**	**01**	**0**	**04**

Course Overview:

The course presents the concepts necessary to understand the way works as well as the system's most commonly used commands. Data manipulation utilities and socket syntax for synthesizing command pipelines are emphasized. Comprehensive hands on exercises are integrated throughout to reinforce learning and develop real competency

Course Objectives:

CO1: To understand the network topology and designing principle.
CO2: Interpreting the client server communication standard protocol and syntaxes.
CO3: Analysing and evaluating the TCP and echo server impact in multiplexing.
CO4: Understanding and designing the new file locking approach for reliable IPC .

CO-PO Mapping:

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	2	3	1	2	1	-	1	-	-	-	1	1
CO2	-	2	2	2	2	-	-	-	1	-	1	1
CO3	-	2	1	2	2	1	1	-	1	-	1	1
CO4	1	2	1	2	2	1	-	1	-	1	-	1

Course Outcomes:

6. Ability to understand and reason out the working of network Systems.
7. To teach students the use of basic socket programming Utilities.
8. To teach students the principles of socket programming
9. To familiarize students with the concepts, design, and structure of the TCP/UDP programming.
10. To be able to build an application of UNIX programming in socket.

Unit-I :

Introduction to Network Programming:

OSI model, Unix standards, TCP and UDP & TCP connection establishment and Format, Buffer sizes and limitation, standard internet services, Protocol usage by common internet application.

Outcome:

2. Ability to understand and reason out the working of Unix base socket programming.

Activity/Event :

Seminar and Class test

Unit-II:

TCP client server:

Introduction, TCP Echo server functions, Normal startup, terminate and signal handling server process termination, Crashing and Rebooting of server host shutdown of server host.

Outcome:

2. To create practical concept of TCP socket, echo server and IPV6 protocol implementation.

Activity/Event :

Seminar and Class test

Unit-III:

Sockets:

Address structures, value – result arguments, Byte ordering and manipulation function and related functions Elementary TCP sockets – Socket, connect, bind, listen, accept, fork and exec function, concurrent servers. Close function and related function. **I/O Multiplexing and socket options**: I/O Models, select function, Batch input, shutdown function, poll function, TCP Echo server, getsockopt and setsockopt functions. Socket states, Generic socket option IPV6 socket option ICMPV6 socket option IPV6 socket option and TCP socket options.

Outcome:

- Student will able to implement socket base programming and design customize server.

Activity/Event :

Seminar and Class test

Unit-IV:

Elementary UDP sockets:

Introduction UDP Echo server function, lost datagram, summary of UDP example, Lack of flow control with UDP, determining outgoing interface with UDP. Elementary name and Address conversions: DNS, gethost by Name function, Resolver option, Function and IPV6 support, uname function, other networking information.

Outcome:

- To familiarize students with the concepts, design, and structure of the UDP sockets and serever with different type of implementation with DNS.

Activity/Event :

Seminar and Class test

Unit-V:

IPC :

Introduction, File and record locking, Pipes, FIFOs streams and messages, Name spaces, system IPC, Message queues, Semaphores. Remote Login: Terminal line disciplines, Pseudo Terminals, Terminal modes, Control Terminals, rlogin Overview, RPC Transparency Issues.

Outcome:

- Student will able to understand the concept of files in network programming and also learn how to manage the synchronization between the terminals.

Activity/Event :

Seminar and Class test

Text Books:

3. Unix and shell Programming, N B Venkateswarlu, Reem
4. Unix Concepts and Applications, 3/e, Sumitabha Das, TMH

Reference Books:

7. UNIX Systems Programming using C++ T CHAN, PHI.
8. UNIX for Programmers and Users, 3rd Edition Graham GLASS, King abls, Pearson Education
9. Advanced UNIX Programming 2nd Edition M. J. ROCHKIND, Pearson Education

Subject Code 4010173503	Python Programming	L 4	T 0	P 0	Credits 3

Course Overview:

- The course is designed to provide an introduction to the python programming language. The focus of the course is to provide students with an introduction to programming, I/O, functions, visualization using the Python programming language.

Course Objectives:

- To understand why Python is a useful scripting language for developers.
- To learn how to design and program Python applications.
- To learn how to use lists, tuples, and dictionaries in Python programs.
- To learn how to use indexing and slicing to access data in Python programs.
- To define the structure and components of a Python program.
- To learn how to write loops and decision statements in Python.
- To learn how to write functions and pass arguments in Python.
- To learn how to build and package Python modules for reusability.
- To acquire object oriented skills in Python.
- To learn how to use exception handling in Python applications for error handling.
- To develop the ability to write database applications in Python

Course Outcomes:

After completion of course students will be able to

CO1	Construct Software easily right out of the box
CO2	Experiment with an interpreted Language
CO3	Build software for real needs
CO4	Explain to testing Orielly

CO-PO Mapping:

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	1	2	-	1	1	-	2	-	-	1	-	1
CO2	2	2	-	1	2	-	1	-	1	-	-	-
CO3	1	1	2	3	1	2	1	1	1	1	1	3
CO4	-	1	-	1	-	2	-	1	-	1	1	2

Unit-I:

Introduction

History of Python, Need of Python Programming, Applications Basics of Python, Programming Using the REPL(Shell), Running Python Scripts, Variables, Assignment, Keywords, Input-Output, Indentation.

Outcome:

- Explain need of python programming
- Explain basic principles of Python Programming language

Activity/Event :

Seminar and Class test

Unit-II:

Types, Operators and Expressions

Types - Integers, Strings, Booleans; Operators- Arithmetic Operators, Comparison (Relational) Operators, Assignment Operators, Logical Operators, Bitwise Operators, Membership Operators, Identity Operators, Expressions and order of evaluations Control Flow- if, if-elif-else, for, while, break, continue, pass.

Outcome:

- Explain the operators in Python
- Explain the control structures in Python

Activity/Event :

Seminar and Class test

Unit-III:

Data Structures

Lists - Operations, Slicing, Methods; Tuples, Sets, Dictionaries, Sequences, Comprehensions.

Outcome:

- Explain the different data structures like list, tuple, dictionary and sequences etc.

Activity/Event :

Seminar and Class test

Unit-IV:

Functions

Defining Functions, Calling Functions, Passing Arguments, Keyword Arguments, Default Arguments, Variable-length arguments, Anonymous Functions, Fruitful, Functions(Function Returning Values), Scope of the Variables in a Function - Global and Local variables.

Modules:

Creating modules, import statement, from. Import statement, name spacing.

Python packages

Introduction to PIP, Installing Packages via PIP, Using Python Packages.

Outcome

- Gains the knowledge in functions and modules
- Implement packages

Activity/Event :

Seminar and Class test

Unit-V:

Object Oriented Programming OOP in Python:

Classes, 'self variable', Methods, Constructor, Method, Inheritance, Overriding Methods, Data hiding.

Error and Exceptions:

Difference between an error and Exception, Handling Exception, try, except block, Raising Exceptions, User Defined Exceptions

Brief Tour of the Standard Library:

Operating System Interface - String Pattern Matching, Mathematics, Internet Access, Dates and Times, Data Compression, Multithreading, GUI Programming, Turtle Graphics.

Introduction to Relational databases:

An introduction to relational databases, SQL statements for data manipulation, creating a GUI that handles an event, working with components, Illustrative programs.

Outcome:

- Implement object oriented concepts
- Implement databases and GUI applications

Activity/Event :

Seminar and Class test

Text Books:

1. Python Programming: A Modern Approach, Vamsi Kurama, Pearson

2. Learning Python, Mark Lutz, Orielly

Reference Books:

1. Think Python, Allen Downey, Green Tea Press

2. Core Python Programming, W. Chun, Pearson.

3. Introduction to Python, Kenneth A. Lambert, Cengag

Subject Code 4010173504	Cyber Security (Elective-III)	L 04	T 01	P 0	Credits 04

Course Overview:

In today's cyber world, it is important for engineers to understand and appreciate computer/information security as it has becomes an essential aspect of our day life. This course provides students with concepts of computer security, cryptography, digital money, secure protocols, detection and other security techniques. Upon the completion of this course, students should be able to understand, appreciate, employ, design and implement appropriate security technologies and policies to protect computers and digital information.

.

Course Objectives:

Cyber Security program will be able to evaluate the computer network and information security needs of an organization.

Assess cyber security risk management policies in order to adequately protect an organization's critical information and assets

Course Outcomes:

- Explore various security policies and evolution of security.
- Investigate more on various catalog approaches and cyber security objectives.
- Analyze cyber user and conflict issues.
- Review cyber management and infrastructure issues.
- Examine various case studies on cyber security policies.

CO-PO Mapping:

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	2									2
CO2			3	2								
CO3		3	3			3						2
CO4		2	3		3							
CO5		2				3						

Unit-I:

Introduction to Cybercrime:

Introduction, Cybercrime: Definition and Origins of the Word, Cybercrime and Information Security, Who are Cybercriminals? , Classifications of Cybercrimes, Cybercrime: The Legal Perspectives, Cybercrimes: An Indian Perspective, Cybercrime and the Indian ITA 2000, A Global Perspective on Cybercrimes, Cybercrime Era: Survival Mantra for the Netizens.

Outcome:

- Outline the origin of different type of threats and risk.
- An ability to analyze a problem, and to identify and define the computing requirements appropriate to its solution.
- An ability to design, implement and evaluate a computer-based solution to meet a given set of computing requirements in the context of the discipline.

Activity/Event :

Seminar and Class Test

Unit-II:

Cyber offenses:

How Criminals Plan Them –Introduction, How Criminals Plan the Attacks, Social Engineering, Cyber stalking, Cyber cafe and Cybercrimes, Botnets: The Fuel for Cybercrime, Attack Vector Cloud Computing.

Outcome:

- An ability to communicate effectively with a range of audiences about technical information.
- An ability to make informed judgments in computing practice based on legal and ethical principles.

Activity/Event :

Seminar and Class Test

Unit-III:

Cybercrime Mobile and Wireless Devices:

Introduction, Proliferation of Mobile and Wireless Devices, Trends in Mobility, Credit Card Frauds in Mobile and Wireless Computing Era, Security Challenges Posed by Mobile Devices, Registry Settings for Mobile Devices, Authentication Service Security, Attacks on Mobile/Cell Phones, Mobile Devices: Security Implications for Organizations, Organizational Measures for Handling Mobile, Organizational Security Policies and Measures in Mobile Computing Era, Laptops.

Outcome:

- An ability to function effectively on teams to establish goals, plan tasks, meet deadlines, manage risk and produce deliverables.
- An ability to apply security principles and practices to the environment, hardware, software, and human aspects of a system

Activity/Event :

Seminar and Class Test

Unit-IV :

Tools and Methods Used in Cybercrime:

Introduction, Proxy Servers and Anonymizers, Phishing, Password Cracking, Key loggers and Spywares, Virus and Worms, Trojan Horses and Backdoors, Steganography, DoS and DDoS Attacks, SQL Injection, Buffer Overflow, Attacks on Wireless Networks, Phishing and Identity Theft: Introduction, Phishing, Identity Theft (ID Theft).

Outcome:

- An ability to function effectively on teams to establish goals, plan tasks, meet deadlines, manage risk and produce deliverables.
- An ability to apply security principles and practices to the environment, hardware, software, and human aspects of a system.
- To analyze and produce different type of tools to handle threats .

Activity/Event :

Unit-V:

Cybercrimes and Cyber Security:

Why Do We Need Cyber laws: The Indian Context, The Indian IT Act, Challenges to Indian Law and Cybercrime Scenario in India, Consequences of Not Addressing the Weakness in Information Technology Act, Digital Signatures and the Indian IT Act, Information Security Planning and Governance, Information Security Policy Standards, Practices, The information Security Blueprint, Security education, Training and awareness program, Continuing Strategies.

Outcome:

- An ability to analyze and evaluate systems with respect to maintaining operations in the presence of risks and threats.

Activity/Event:

Seminar and Class Test

Text Books:

1. Cyber Security: Understanding Cyber Crimes, Computer Forensics and Legal Perspectives, Nina Godbole, SunitBelapure, Wiley.

2. Principles of Information Security, MichealE.Whitman and Herbert J.Mattord, Cengage Learning

Reference Books:

1. Information Security, Mark Rhodes, Ousley, MGH.

Subject Code	Computer Forensics (Elective-III)	L	T	P	Credits
4010173505		04	01	0	04

Course Overview:

Computer forensics (also known as computer forensic science is a branch of digital forensic science pertaining to evidence found in computers and digital storage media. The goal of computer forensics is to examine digital media in a forensically sound manner with the aim of identifying, preserving, recovering, analyzing and presenting facts and opinions about the digital information. Although it is most often associated with the investigation of a wide variety of computer crime, computer forensics may also be used in civil proceedings. The discipline involves similar techniques and principles to data recovery, but with additional guidelines and practices designed to create a legal audit trail**.**

Course Objectives:

Forensic investigators typically follow a standard set of procedures and maintain physically isolating the devices in question to make sure it cannot be accidentally contaminated, investigators make a digital copy of the device's storage media. Once the original media has been copied, it is locked in a safe or other secure facility to maintain its pristine condition. All investigation is done on the digital copy.

Course Outcomes:

CO1: Explain the role of forensics in preventing various forms of fraud.
CO2: Develop skills in distinguishing various types of computer crimes and identify the digital fingerprints associated with criminal activities.
CO3: Illustrate how to apply different forensic analysis tools to recover important evidence for identifying computer crimes.
CO4: Explain about threats and compare various threats.
CO5: Summarize the need for surveillance and list the tools used.

CO-PO Mapping:

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	2			2								
CO2						2						
CO3				2	1							
CO4	2	2										
CO5	2											

Unit-I:

Computer Forensics and Investigations:

Understanding Computer Forensics, Preparing for
Computer Investigations, Taking A Systematic Approach, Procedure for Corporate High-Tech Investigations, Understanding Data Recovery Workstations and Software,

Investor's Office and Laboratory:

Understanding Forensics Lab Certification Requirements, Determining the Physical Requirements for a Computer Forensics Lab, Selecting a Basic Forensic Workstation

Outcome:

Student will able to understand the basic concept of computer forensics and investigations.

Activity/Event :

Seminar and Class Test

Unit-II:

Data Acquisition: Understanding Storage Formats for Digital Evidence, Determining the Best Acquisition Method, Contingency Planning for Image Acquisitions, Using Acquisition Tools, Validating Data Acquisition, Performing RAID Data Acquisition, Using Remote Network Acquisition Tools, Using Other Forensics Acquisition Tools

Processing Crime and Incident Scenes:

Identifying Digital Evidence, Collecting the Evidence in Private-Sector Incident Scenes, Processing law Enforcement Crime Scenes, Preparing for a Search, Securing a Computer Incident or Crime Scene, Sizing Digital evidence at the Scene, Storing Digital evidence, obtaining a Digital Hash.

Unit-II Outcome:

Student will able to understand Data Acquisition and how to identify digital evidences and processing law enforcement.

Activity/Event :

Seminar and Class Test

Unit-III:

Current Computer Forensics Tools:

Evaluating Computer Forensics Toll Needs, Computer
Forensics Software Tools, Computer Forensics Hardware Tools, Validating and Testing Forensics Software

Computer Forensics Analysis and Validation:

Determining What Data to Collect and Analyze, Validating Forensic Data, Addressing Data-Hiding Techniques, Performing Remote Acquisition..

Outcome:

Student will able to understand the computer forensics tools and how to do analysis and validation.

Activity/Event :

Seminar and Class Test

Unit-IV:

Recovering Graphics and Network Forensics:

Recognizing a Graphics File, Understanding
Data Compression, Locating and Recovering Graphics Files, Understanding Copyright Issues with Gra
Network Tools, Examining Hiney Project.

Outcome:

Students will able to understand how to recover.

Activity/Event :

Seminar and Class Test

Unit-V:

E-mail Investigations Cell Phone and Mobile Device Forensics:

Exploring the Role of E-mail
in Investigations, Exploring the Role of Client and Server in E-mail, Investigating E-mail Crimes and Violations, Understanding E-mail Servers, Using Specialized E-mail Forensics Tools, Understanding Mobile Device Forensics, Understanding Acquisition Procedure for Cell Phones and Mobile Devices.

Outcome:

Students will be able to investigate cell phone and mobile device forensics using tools.

Activity/Event :

Seminar and Class Test

Text Books:

1. Nelson, Phillips Enfinger,Steuart, " Computer Forensics and Investigations, Cengage Learning.

Reference Books:

1. Computer Forensics and Cyber Crime: An introduction, 2e by Britz

Subject Code	Ecommerce (Elective-III)	L	T	P	Credits
4010173506		4	0	0	4

Course Overview:

Students should have knowledge of

The concepts, vocabulary, and procedures associated with E-Commerce and the Internet. The student gains an overview of all aspects of E-Commerce. Topics include development of the Internet and E-Commerce, options available for doing business on the Internet, features of Web sites and the tools used to build an E-Commerce web site, marketing issues, payment options, security issues, and customer service..

Course Objectives:

The objectives of the course are:

- To understand the interest and opportunity of e-commerce.
- To know and understand the critical success factors in implementing an ecommerce system.
- To know how to plan and how to manage e-commerce solutions.
- To apply processes of e-commerce.
- To analyze and understand the human, technological and business environment associated with e-commerce.
- To know how to use technologies to build e-commerce websites.

Course Outcomes:

After completion of course students will be able to
CO1: Gain an understanding of the theories and concepts underlying e-commerce.
CO2: Apply e-commerce theory and concepts to what e-marketers are doing in "the real world"
CO3: Review e-Commerce infrastructures including architecture models, security & payment systems.
CO4: Improve familiarity with current challenges and issues in e-commerce.
CO5: Identify business models surrounding e-Commerce including marketing strategies.

CO-PO Mapping:

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	2				2					1		
CO2			2	2		1					1	
CO3	2								2			
CO4		1										1
CO5					1					2		

Unit-I:

Electronic Commerce, Frame work, anatomy of E-Commerce applications, E-Commerce Consumer applications, E-Commerce organization applications. Consumer Oriented Electronic commerce, Mercantile Process models.

Outcome:

- Demonstrate an understanding of the foundations and importance of E-commerce
- Demonstrate an understanding of retailing in E-commerce by analyzing branding and pricing strategies.
- Demonstrate an understanding of retailing in E-commerce by using and determining the effectiveness of market research.
- Demonstrate an understanding of retailing in E-commerce by assessing the effects of disintermediation.

Activity/Event :

Group Discussion

Unit-II:

Electronic payment systems - Digital Token-Based, Smart Cards, Credit Cards, Risks in Electronic Payment systems.

Outcome:

- Analyze the impact of E-commerce on business models and strategy
- Describe Internet trading relationships including Business to Consumer, Business-to-Business, Intra-organizational.
- Describe the infrastructure for E-commerce
- Describe the key features of Internet, Intranets and Extranets and explain how they relate to each other.

Activity/Event :

Test / Seminar

Unit-III:

Inter Organizational Commerce - EDI, EDI Implementation, Value added networks.
Intra Organizational Commerce - work Flow, Automation Customization and internal Commerce, Supply chain Management.

Outcome:

- Discuss legal issues and privacy in E-Commerce
- Assess electronic payment systems
- Recognize and discuss global E-commerce issues

Activity/Event :

Group Discussion/ Seminar

Unit-IV:

Corporate Digital Library - Document Library, digital Document types, corporate Data Warehouses.
Advertising and Marketing, Information based marketing, Advertising on Internet, on-line marketing process, market research.

Outcome:

- Describe the infrastructure for E-commerce
- Describe the key features of Internet, Intranets and Extranets and explain how they relate to each other.
- Discuss legal issues and privacy in E-Commerce
- Assess electronic payment systems

Activity/Event :

Test/ Seminar

Unit-V:

Consumer Search and Resource Discovery, Information search and Retrieval, Commerce Catalogues, Information Filtering. Multimedia - key multimedia concepts, Digital Video and electronic Commerce, Desktop video processings, Desktop video conferencing.

Outcome:

- Assess electronic payment systems
- Recognize and discuss global E-commerce issues.
- Demonstrate an understanding of retailing in E-commerce by assessing the effects of disintermediation.

Activity/Event :

Test/ Seminar

Text Books:

1. Frontiers of Electronic Commerce , Kalakata, Whinston, PEA,2006.

Reference Books:

1. E-Commerce Fundamentals and Applications Hendry Chan, Raymond Lee, Dillon, Chang, John Wiley.
2. E-Commerce, A Managerial Perspective, Turban E, Lee J , King, Chung H.M.,PEA,2001.
3. E-Commerce An Indian Perspective , 3/e, P.T. Joseph, PHI,2009.
4. E-Commerce, S. Jaiswal, Galgotia.
5. Electronic Commerce , Gary P. Schneider, Thomson.

Subject Code	Internet of Things (Elective-IV)	L	T	P	Credits
4010173507		4	0	0	3

Course Overview:

The course covers popular, service-rich cloud platforms and focuses on how to build and deploy IoT solutions. Practical use cases and case studies are included to ensure that the candidate develops an ability to work through practical real-life scenarios.

Course Objectives:

Students will be explored to the interconnection and integration of the physical world and the cyber space. They are also able to design &develop IOT Devices.

Course Outcomes:

- Demonstrate knowledge and understanding of the security and ethical issues of the Internet of Things
- Conceptually identify vulnerabilities ,including recent attacks ,involving the Internet of Things
- Develop critical thinking skills
- Compare and contrast the threat environment based on Industry or device type.

CO-PO Mapping:

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO 10	PO 11	PO 12
CO1	1	2	1	2	1	-	-	-	-	-	1	1
CO2	-	2	2	2	2	-	-	-	-	-	1	1
CO3	-	2	2	2	3	1	-	-	-	-	1	1
CO4	1	3	1	2	3	1	-	-	-	-	-	-

Unit-I:

The Internet of Things:

An Overview of Internet of things, Internet of Things Technology, behind Io Ts Sources of the Io Ts, M2M Communication, Examples OF Io Ts, Design Principles For Connected Devices Internet Connectivity Principles, Internet connectivity, Application Layer Protocols: HTTP, HTTPS, FTP, Telnet.

Outcome:

Sound understanding of core concepts, background technologies, and sub-domains of IoT

Activity/Event :

Hands on experience on various IOT devices

Unit-II:

Business Models for Business Processes in the Internet of Things ,IoT/M2M systems LAYERS AND designs standardizations ,Modified OSI Stack for the IoT/M2M Systems ,ETSI M2M domains and High-level capabilities ,Communication Technologies, Data Enrichment and Consolidation and Device Management Gateway Ease of designing and affordability.

Outcome:

- Sensors, microcontrollers and communication interfaces to design and build IoT devices

Activity/Event :

Seminar and class test

Unit-III:

Design Principles for the Web Connectivity for connected-Devices, Web Communication protocols for Connected Devices, Message Communication protocols for Connected Devices, Web Connectivity for connected-Devices.

Outcome:

Analysis and evaluate protocols used in IOT

Activity/Event :

Practicing real time applications

Unit-IV:

Data Acquiring, Organizing and Analytics in IoT/M2M, Applications/Services/Business Processes, IOT/M2M Data Acquiring and Storage, Business Models for Business Processes in the Internet Of Things, Organizing Data, Transactions, Business Processes, Integration and Enterprise Systems.

Outcome:

Analysis and Evaluate the data received through sensors in IOT

Activity/Event:

Programming Contest using Python programming

Unit-V:

Data Collection, Storage and Computing Using a Cloud Platform for IoT/M2M Applications/Services, Data Collection, Storage and Computing Using cloud platform Everything as a service and Cloud Service Models, IOT cloud-based services using the Xively (Pachube/COSM), Nimbits and other platforms Sensor, Participatory Sensing, Actuator, Radio Frequency Identification, and Wireless, Sensor Network Technology, Sensors Technology, Sensing the World.

Outcome:

Apply IOT to different applications

Activity/Event :

Practicing examples

Text Books:

1Internet of Things: Architecture, Design Principles And Applications, Rajkamal, McGraw Hill

Higher Education

2.Internet of Things, A. Bahgya and V. Madisetti, University Press, 2015

Reference Books:

1. Designing the Internet of Things, Adrian McEwen and Hakim Cassimally, Wiley
2. Getting Started with the Internet of Things CunoPfister , Oreilly.

Subject Code 4010173507	MMAD (Elective-IV)	L 4	T 0	P 0	Credits 4

Course Overview:

Students should have knowledge of

- Formulate a working definition of interactive multimedia.
- Demonstrate competence in using the authoring program Hyper Studio.
- Demonstrate the use of animation, digitized sound, video control, and scanned images.
- Demonstrate the use of Netscape to access the Course Home Page and Tips and Tricks.
- Use basic instructional design principles in the development of stacks.
- Will develop conceptual maps of content and process for interactive multimedia instructional programs.

Course Objectives:

The objectives of the course are:

- To learn and understand technical aspect of Multimedia Systems.
- To understand the standards available for different audio, video and text applications.
- To Design and develop various Multimedia Systems applicable in real time.
- To learn various multimedia authoring systems.
- To understand various networking aspects used for multimedia applications.
- To develop multimedia application and analyze the performance of the same.

Course Outcomes:

After completion of course students will be able to:

- Developed understanding of technical aspect of Multimedia Systems.
- Understand various file formats for audio, video and text media.
- Develop various Multimedia Systems applicable in real time.
- Design interactive multimedia software.
- Apply various networking protocols for multimedia applications.

CO-PO Mapping:

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3			3								1
CO2		2	1								2	
CO3	2		2		2							2
CO4		2	2								2	
CO5	1			2					1			

Unit - I:

Fundamental concepts in Text and Image:

Multimedia and hypermedia, world wide web, overview of multimedia software tools. Graphics
and image data representation graphics/image data types, file formats, Color in image and video:
color science, color models in images, color models in video

Outcome:

- Developed understanding of technical aspect of Multimedia Systems.
- Understand various file formats for audio, video and text media.
- Develop various Multimedia Systems applicable in real time.

Activity/Event:

Group Discussion

Unit-II :

Fundamental Concepts in Video and Digital Audio:

Types of video signals, analog video, digital video, digitization of sound, MIDI, quantization
And transmission of audio.

Outcome:

- Demonstrate the use of animation, digitized sound, video control, and scanned images.
- Demonstrate the use of Netscape to access the Course Home Page and Tips and Tricks.
- Use basic instructional design principles in the development of stacks.

Activity/Event:

Test / Seminar/Programming Contest

Unit-III:

Action Script I:

Action Script Features, Object-Oriented Action Script, Data types and Type Checking, Classes, Authoring an Action Script Class.

Action Script II:

Inheritance, Authoring an Action Script 2.0 Subclass, Interfaces, Packages, Exceptions.

Application Development:

An OOP Application Frame work, Using Components with Action Script Movie Clip Subclasses.

Outcome:

- Learn various multimedia authoring systems.
- Understand various networking aspects used for multimedia applications.
- Develop multimedia application and analyze the performance of the same.

Activity/Event:

Practicing real time applications

Unit-IV:

Multimedia Data Compression:

Lossless compression algorithm: Run-Length Coding, Variable Length Coding, Dictionary Based Coding, Arithmetic Coding, Lossless Image Compression, Lossy compression algorithm:
Quantization, Transform Coding, Wavelet-Based Coding, Embedded Zero tree of Wavelet Coefficients Set Partitioning in Hierarchical Trees (SPIHT).

Basic Video Compression Techniques:

Introduction to video compression, video compression based on motion compensation, search
for motion vectors, MPEG, Basic Audio Compression Techniques.

Outcome:

- Design interactive multimedia software.
- Apply various networking protocols for multimedia applications.
- To evaluate multimedia application for its optimum performance

Activity/Event :

Programming Contest

Unit-V:

Multimedia Networks:

Basics of Multimedia Networks, Multimedia Network Communications and Applications Quality of Multimedia Data Transmission, Multimedia over IP, Multimedia over ATM networks, Transport of MPEG- 4, Media-on- Demand (MOD).

Outcome:

- To understand the standards available for different audio, video and text applications.
- To Design and develop various Multimedia Systems applicable in real time.
- To learn various multimedia authoring systems.
- To understand various networking aspects used for multimedia applications.

Activity/Event :

Practicing examples

Text Books:

1. Fundamentals of Multimedia , Ze-Nian Li , Mark S. Drew, PHI/ PEA.

2. Multimedia Systems, Parag Havaldar, Gerard Medioni, cengage, 2009.

3. EssentialsAction Script 3.0, Colin Moock, SPD O, Reilly, 2007.

Reference Books:

1. Multimedia Applications, Steinmetz, Nahrstedt, Springer.

2. Digital Multimedia, Nigel Chapman, Jenny Chapman, Wiley- Dreamtech.

3. Multimedia & Communications Technology, Steve Heath, Elsevier .

4. Multimedia Technology & Applications, David Hilman , Galgotia.

5. Multimedia Technologies, Banerji, Mohan Ghosh, MGH.

Subject Code	**SOFTWARE TESTING METHODOLOGIES (Elective-IV)**	**L**	**T**	**P**	**Credits**
4010173509		**4**	**0**	**0**	**3**

Course Overview:

To study the fundamental concepts of software testing which includes objectives, process, criteria strategies, and methods, discuss various software testing types and levels of testing like black and white box.

Course Objectives:

- Understand the importance of testing and debugging.
- Interpret a model for testing and understand the process of testing.
- Discuss the limitations of testing.
- Identify the consequences and understand the importance of bugs.
- Classify the bugs into different categories.

Course Outcomes:

1. Analyze the Conventional Software Management and improving Software Economics.
2.Demonstrate the principles of conventional software Engineering, Life cycle Phases, and Artifacts of the process.
3.Apply the Software testing Work Flows of the process, Checkpoints of the process and Iterative Process Planning.
4.Develop automation Process, Project Control and Process instrumentation, tailoring the process in software testing.
5. Evaluate the project organizations and responsibilities, future software project management with case study

CO-PO Mapping:

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1								3	3			
CO2	2	3							3			
CO3			3	3					2			
CO4				2	3							
CO5					2	3	3					

Unit-I:

Introduction:-

Purpose of testing, Dichotomies, model for testing, consequences of bugs, taxonomy of bugs Flow graphs and Path testing:- Basics concepts of path testing, predicates, path predicates and achievable paths, path sensitizing, path instrumentation, application of path testing.

Outcome:

- Have an ability to apply software testing knowledge and engineering methods.
- To apply the fundamental knowledge of testing real time scenarios

Activity/Event :

Seminar

Unit-II:

Transaction Flow Testing:-transaction flows, transaction flow testing techniques.
Dataflow testing:- Basics of dataflow testing, strategies in dataflow testing, application of dataflow testing

Outcome:

To test a simple application of their choice and to understand those learnt techniques in software development life cycle. Have an ability to design and conduct a software test process for a software

Activity/Event :

Seminar/ Class Test

Unit-III:

Domain Testing:-domains and paths, Nice & ugly domains, domain testing, domains and interfaces testing, domain and interface testing, domains and testability.

Outcome:

Have an ability understand and identify various software testing problems, and solve these problems by designing and selecting software test models

Activity/Event :

Practicing real time applications

Unit-IV:

Paths, Path products and Regular expressions:- path products & path _expression, reduction procedure, applications, regular expressions & flow anomaly detection. Logic Based Testing:- overview, decision tables, path expressions, kv charts, specifications.

Outcome:

Have basic understanding and knowledge of contemporary issues in software testing, such as component based software testing problems.

Activity/Event:

Group discussion

Unit-V:

State, State Graphs and Transition testing:- state graphs, good & bad state graphs, state testing, Testability tips. Graph Matrices and Application:-Motivational overview, matrix of graph, relations, power of a matrix, node reduction algorithm, building tools.

Outcome:

Have an ability to use software testing methods and modern software testing tools for their testing projects

Activity/Event :

Practicing examples

Text Books:

1. Software testing techniques - Boris Beizer, International Thomson computer press, second edition.
2. Software Testing- Yogesh Singh, CAMBRIDGE

Reference Books:

1. Introduction to Software Testing, Paul Amman, Jeff Offutt, CAMBRIDGE
2. Effective Software testing, 50 Specific ways to improve your testing, Elfriede Dustin, PEA

Subject Code	BIG DATA ANALYTICS LAB	L	T	P	Credits
4010173521		0	0	3	2

Course Overview:

- The main objective of this lab is to impart the knowledge on how to Store, manage, and analyze unstructured data and Query large data sets in near real time with Pig and Hive

Course Objectives:

5. Students will learn the basics of Hadoop
6. The main objective of this lab is to impart the knowledge on how to implement concept and challenge of big data (3 V's: volume, velocity, and variety).
7. Teach students in applying skills and tools to manage and analyze the big data.

Course Outcomes:

- To understand the basic principles, concepts of Big Data Analyze and interpret data using an ethically responsible approach.
- Collect, manage, store, query, and analyze various form of big data
- Gain hands-on experience on large-scale analytics tools to solve some open big data problems
- Understand the impact of big data for business decisions and strategy.

Week 1, 2:

1. Implement the following Data structures in Java
a)Linked Lists b) Stacks c) Queues d) Set e) Map

Week 3, 4:

2. (i)Perform setting up and Installing Hadoop in its three operating modes:
Standalone, Pseudo distributed, Fully distributed
(ii)Use web based tools to monitor your Hadoop setup.

Week 5:

3. Implement the following file management tasks in Hadoop:
- Adding files and directories
- Retrieving files
- Deleting files

Hint: A typical Hadoop workflow creates data files (such as log files) elsewhere and copies them into HDFS using one of the above command line utilities.

Week 6:

4. Run a basic Word Count Map Reduce program to understand Map Reduce Paradigm.

Week 7:

5. Write a Map Reduce program that mines weather data.
Weather sensors collecting data every hour at many locations across the globe gather a large volume of log data, which is a good candidate for analysis with MapReduce, since it is semi structured and record-oriented.

Week 8:

6. Implement Matrix Multiplication with Hadoop Map Reduce

Week 9, 10:

7. Install and Run Pig then write Pig Latin scripts to sort, group, join, project, and filter your data.

Week 11, 12:

8. Install and Run Hive then use Hive to create, alter, and drop databases, tables, views, functions, and indexes

Reference Books:

1. Hadoop in Practice by Alex Holmes, MANNING Publ.
2. Hadoop MapReduce Cookbook, Srinath Perera, Thilina Gunarathne.

Subject Code	NETWORK PROGRAMMING LAB	L	T	P	Credits
4010173522		0	0	3	2

Week 1:

1.Write a shell script that accepts a file name, starting and ending line numbers as arguments and displays all the lines between the given line numbers

Week 2:

2.Write a shell script that deletes all lines containing a specified word in one or more files supplied

as arguments to it.

Week 3:

3. Write a shell script that displays a list of all the files in the current directory to which the user has read, write and execute permissions.

Week 4:

4. Write a shell script that receives any number of file names as arguments checks if every argument supplied is a file or a directory and reports accordingly. Whenever the argument is a file, the number of lines on it is also reported.

Week 5:

5. Write a shell script that accepts a list of file names as its arguments, counts and reports the occurrence of each word that is present in the first argument file on other argument files.

Week 6:

11. Write a shell script that accepts any number of arguments and prints them in the reverse order.

Week 7:

12. Write a shell script that determines the period for which a specified user is working on the system.

Week 8:

8. Write a shell script to list all of the directory files in a directory

Week 9:

9. Write an interactive file-handling shell program- Let it offer the user the choice of copying, removing or linking files. Once the user has made a choice, have the program ask him for the necessary information such as the file name, new name and so on.

Week 10:

10. Write a shell script to find factorial of a given integer.

Week 11:

11. Write a shell script to find the G.C.D. of two integers

Week 12:

12. Write a shell script to generate a multiplication table.

Week 13:

13. Write a shell script that copies multiple files to a directory.

Week 14:

14. Write a shell script that counts the number of lines and words present in a given file.

Subject Code	PYTHON PROGRAMMING LAB	L	T	P	Credits
4010173523		0	0	3	2

Detailed Syllabus:

Week 1: Exercise 1 – Basics

a. Running instructions in Interactive interpreter and a Python Script

b. Write a program to purposefully raise Indentation Error and Correct it

Week 2: Exercise 2 – Operations

a. Write a program to compute distance between two points taking input from the user (Pythagorean Theorem)

b. Write a program add.py that takes 2 numbers as command line arguments and prints its sum.

Week 3: Exercise - 3 Control Flow

a. Write a Program for checking whether the given number is a even number or not.

b. Using a for loop, write a program that prints out the decimal equivalents of 1/2, 1/3, 1/4, . . . , 1/10

c. Write a program using a for loop that loops over a sequence. What is sequence?

d. Write a program using a while loop that asks the user for a number, and prints a countdown from that number to zero.

Week 4: Exercise 4 - Control Flow – Continued

a. Find the sum of all the primes below two million.

b. Each new term in the Fibonacci sequence is generated by adding the previous two terms. By starting with 1 and 2, the first 10 terms will be:
1, 2, 3, 5, 8, 13, 21, 34, 55, 89, ...

c. By considering the terms in the Fibonacci sequence whose values do not exceed four million, find the sum of the even-valued terms.

Week 5: Exercise - 5 - DS

a. Write a program to count the numbers of characters in the string and store them in a dictionary data structure
b. Write a program to use split and join methods in the string and trace a birthday with a dictionary data structure.

Week 6: Exercise - 6 DS - Continued

a. Write a program combine_lists that combines these lists into a dictionary.
b. Write a program to count frequency of characters in a given file. Can you use character frequency to tell whether the given file is a Python program file, C program file or a text file?

Week 7: Exercise – 7 Files

a. Write a program to print each line of a file in reverse order.
b. Write a program to compute the number of characters, words and lines in a file.

Week 8: Exercise - 8 Functions

a. Write a function ball_collide that takes two balls as parameters and computes if they are colliding. Your function should return a Boolean representing whether or not the balls are colliding.

Hint: Represent a ball on a plan e as a tuple of (x, y, r), r being the radius
If (distance between two balls centers) <= (sum of their radii) then (they are colliding) b) Find mean, median, mode for the given set of numbers in a list.

Week 9: Exercise - 9 Functions – Continued

a. Write a function nearly_equal to test whether two strings are nearly equal. Tw o strings a and b are nearly equal when a can be generated by a single mutation on b.
b. Write a function dups to find all duplicates in the list.
c. Write a function unique to fin d all the unique elements of a list.

Week 10: Exercise - 10 - Functions - Problem Solving

a. Write a function cumulative product to compute cumulative product of a list of numbers.
b. Write a function reverse to re verse a list. Without using the reverse function.
c. Write function to compute gcd , lcm of two numbers. Each function shouldn’t exceed one line.

Week 11: Exercise 11 - Multi-D Lists

Write a program that defines a matrix and prints

a. Write a program to perform addition of two square matrices
b. Write a program to perform multiplication of two square matrices

Week 12: Exercise - 12 - Modules

a. Install packages requests, fl ask and explore them. using (pip)
b. Write a script that imports requests and fetch content from the page. Eg. (Wiki)
c. Write a simple script that serves a simple HTTP Response and a simple HT ML Page

Week 13: Exercise - 13 OOP

a. Class variables and instance variable and illustration of the self variable

Robot
ATM Machine

Week 14: Exercise - 14 GUI, Graphics

a. Write a GUI for an Expression Calculator using tk
b. Write a program to implement the following figures using turtle

Week 15: Exercise - 15 - Testing

a. Write a test-case to check the function even _numbers which return True on passing a list of all even numbers
b. Write a test-case to check the function reverse _string which returns the reversed string

Week 16: Exercise - 16 – Advanced

a. Build any one classical data structure.
b. Write a program to solve knapsack problem.