ACADEMIC REGULATIONS
COURSE STRUCTURE
AND
DETAILED SYLLABUS

R24

Master of Computer Applications

Two years P.G. Programme.

(Applicable for the batches admitted from the A.Y. 2024-2025)

SWARNANDHRA

COLLEGE OF ENGINEERING & TECHNOLOGY
(AUTONOMOUS)

SEETHARAMAPURAM, NARSAPUR-534 280, W.G.DT., A.P.

ACADEMIC REGULATIONS R24

Master of Computer Applications (MCA) Programme (Duration: Two Years)

(Applicable for the batches admitted from the **A.Y. 2024-25**)

Applicable for the students of Master of Computer Applications (MCA) PG Programme admitted from the Academic Year 2024-25 onwards. The MCA Degree of Jawaharlal Nehru Technological University Kakinada shall be conferred on candidates who are admitted to the program and who fulfil all the requirements for the award of the Degree.

1. ELIGIBILITY FOR ADMISSIONS

Admission to the above program shall be made subject to eligibility, qualification and specialization as prescribed by the University from time to time.

Admissions shall be made on the basis of merit rank obtained by the candidates at ICET examination or the qualifying Entrance Test conducted by the University or on the basis of any other order of merit as approved by the University, subject to reservations as laid down by the Govt. from time to time.

2. AWARD OF MCA DEGREE

2.1 A student shall be declared eligible for the award of the MCA Degree, if he pursues a course of study and completes it successfully in not less than two academic years and not more than Four academic years.

2.2 The student shall register for all 80 credits and secure all the 80 credits.

2.3 The minimum instruction days in each semester are 90.

2.4 A Student, who fails to fulfill all the academic requirements for the award of the degree within Four academic years from the year of their admission, shall forfeit his seat in MCA course.

2.5 Credit Definition:

1 Hour Lecture (L) per week	1 Credit
1 Hour Tutorial (T) per week	1 Credit
1 Hour Practical (P) per week	0.5 Credit

3. ATTENDANCE

3.1 A candidate shall be deemed to have eligibility to write end semester examinations if he has put in a minimum of 75% of attendance in aggregate of all the subjects.

3.2 Condonation of shortage of attendance up to 10% i.e., 65% and above, and below 75% may be given for a **maximum of TWO times** by the college academic committee.

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

3.4 Shortage of attendance below 65 % in aggregate shall not **be condoned and not eligible to write their end semester examination of that class.**

3.5 A candidate shall not be promoted to the next semester unless; he/she fulfills the attendance requirements of the previous semester.

3.6 A stipulated fee of Rs 500/- shall be payable towards condonation fee for shortage of attendance. Students availing condonation on medical ground shall produce a medical certificate issued by the competitive authority.

3.7 A student shall not be promoted to the next semester unless he satisfies the attendance requirement of the present semester, as applicable. They may seek readmission into that semester when offered next. If any candidate fulfills the attendance requirement in the present semester, he shall not be eligible for readmission into the same class.

4. EVALUATION

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

For the theory subjects 60 marks shall be awarded based on the performance in the End Semester Examination and 40 marks shall be awarded based on the Internal Evaluation.

4.1 Continuous Internal Evaluation:

Theory

(a) For theory subjects, during a semester, there shall be two mid-term examinations. Each midterm examination shall be conducted for a total duration of 90 minutes with 4 questions (without choice) each question for 10 marks.

(b) The descriptive examination is set with 4 full questions from first two and half units (50% of the syllabus), the student has to answer all questions. In the similar lines, descriptive examination shall be conducted on the rest of the syllabus.

(c) The first mid (Mid-1) marks shall be submitted to the examination section within one week after completion of first mid examination.

(d) The mid marks submitted to the examination section shall be displayed in the concerned department notice boards for the benefit of the students.

(e) If any discrepancy found in the submitted Mid-1 marks, it shall be brought to the notice of examination section within one week from the submission.

(f) Second mid examination shall be conducted on the similar lines of mid-1 and its mid (Mid-2) marks shall also be submitted to examination section within one week after completion of second mid examination and it shall be displayed in the notice boards. If any discrepancy found in the submitted mid-2 marks, it shall be brought to the notice of examination section within one week from the submission.

(g) The final marks are the sum of average of two mid-term examinations i.e. $\frac{\textbf{Mid1}+\textbf{Mid2}}{\textbf{2}}$

4.2 C End Semester Theory Examination Evaluation:

Theory:

The end semester examinations shall be conducted by the examination section for 60 marks consists of five questions carrying 12 marks each. Each of these questions may contain sub-questions. For each question there will be an “either” “or” choice, which means that there will be two questions from each unit and the student should answer either of the two questions.

4.3 Laboratory Evaluation:

Internal Evaluation: The internal marks for laboratory are 40 marks and the marks shall be awarded based on the day to day work: 10 marks, Record: 5 marks and the remaining 25 marks to be awarded by conducting an internal laboratory test.

External Evaluation: For external marks for laboratory are 60 and marks shall be awarded based on the performance in the end laboratory examinations. Laboratory examination must be conducted with two Examiners, one of them being the Laboratory Class Teacher and the second examiner shall be appointed by the COE from the panel of examiners submitted by the respective college. Laboratory examination must be conducted with a breakup marks of Procedure-15, Experimentation-25, Results-10, Viva-voce-10.

4.4 There shall be an internship / industry oriented mini project/ skill development course, one need to complete during year break (*i.e., II-Sem to III-Sem*) and will be evaluated for 50 marks internally at the end of III Semester by the departmental committee. For skill development course the certificate has to be verified and submitted to the Exam Section. A candidate has to secure a minimum 50% of marks to be declared successful.

4.5 For non-credit Employability Skills, "Satisfactory" or "Unsatisfactory" shall be indicated instead of the letter grade and this will not be counted for the computation of SGPA/CGPA/Percentage

4.6 A candidate shall be deemed to have secured the minimum academic requirement in a subject if he secures a minimum of 40% of marks in the End Semester Examination and a minimum aggregate of 50% of the total marks in the End Semester Examination and Internal Evaluation taken together.

4.7 In case the candidate does not secure the minimum academic requirement in any subject (as specified in 4.6) he has to reappear for the End Semester Examination in that subject. 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 has failed in the end examination. In such a case, the candidate must re-register for the subject(s) and secure the required minimum attendance. The candidate's attendance in the re-registered subject(s) shall be calculated separately to decide upon his eligibility for writing the end examination in those subject(s). In the event of the student taking another chance, his internal marks and end examination marks obtained in the previous attempt are nullified. For re-registration the candidates have to apply to the University through the college by paying the requisite fees and get approval from the University before the start of the semester in which re-registration is required. At a given time, a candidate is permitted to re-register for maximum of two subjects in addition to the subjects of regular semester, when the student is on roles. Once the completion of the course work, a student can re-register a maximum of SIX Courses at any time"

4.8 A Candidate shall be given one chance to re-register for each course provided the internal marks secured by a candidate are less than 50 per cent and he has failed in the end examination after completion of the two years. In such case, the candidate must reregister for the subject(s) and secure required minimum attendance. Attendance in the re-registered subject(s) should be calculated separately to become eligible to write the end examination in the re-registered subject(s). The attendance of re-registered subject(s) shall be calculated separately to decide his eligibility for taking the end examination in those subject(s). 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 two subject(s). For re-registration the candidates have to apply to the Institute by paying the requisite fees and get approval from the concern authorities before the start of the semester in which re-registration is required. In case the candidate secures less than the required attendance in any re-registered course(s), he/she shall not be permitted to write the End Examination in that course.

4.9 In case the candidate secures less than the required attendance in any re registered subject (s), he/she shall not be permitted to write the End Semester Examination in that subject. He shall again re-register the subject when next offered.

4.10 A candidate 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 (4^{th}semester).

5. EVALUATION OF PROJECT/DISSERTATION WORK

Every candidate shall be required to submit a thesis or dissertation on a topic approved by the Project Review Committee.

5.0 Every candidate shall be required to submit thesis or dissertation after taking up a topic approved by the Project Review Committee.

5.1 A Project Review Committee (PRC) shall be constituted with Head of the Department and two other senior faculty members of the concerned department.

5.2 Registration of Project Work: A candidate is permitted to register for the project work after satisfying the attendance requirement of all the courses (theory and practical subjects) up to III semester.

5.3 After satisfying 6.3, a candidate has to submit, in consultation with his project supervisor, the title, objective and plan of action of his project work to the Project Review Committee for its approval. After obtaining the approval of the Committee the student can initiate the Project work after the third semester end examinations.

5.4 Every candidate shall work on projects approved by the PRC of the College

5.5 The duration of the project is for one semester.

5.6 If a candidate wishes to change his supervisor or topic of the project he can do so with approval of the PRC. However, the Project Review Committee (PRC) shall examine whether the change of topic/supervisor leads to a major change of his initial plans of project proposal. If so, his date of registration for the project work starts from the date of change of Supervisor or topic as the case may be.

5.7 A candidate shall submit status report in two stages at least with a gap of one month between them.

5.8 The work on the project shall be initiated in the beginning of the fourth semester and the duration of the project is for one semester. A candidate shall be allowed to submit the project report only with the approval of PRC and not earlier than 16 weeks from the date of registration of the project work. For the approval of PRC the candidate shall submit the draft copy of thesis to the Principal (through Head of the Department) and shall make an oral presentation before the PRC.

5.9 Three copies of the Project Thesis certified by the supervisor & HOD shall be submitted to the College / Department.

5.10 The project work carried out by the candidate during 4^{th} semester is evaluated for internal assessment and external examination.

a) **Internal Assessment**: Internal Assessment will be carried out by the Project Review Committee consisting of

1) Head of the Department
2) Supervisor and
3) Senior faculty members.

Internal Assessment shall be on the basis of two seminars given by the each student on the topic of his project.

b) **External Examination**: External Examination(Viva – Voce) will be conducted by Project External Examination committee consisting of

i. Head of the Department
ii. Supervisor and

iii. External member.

External examiner will be appointed by the COE from the panel of examiners submitted by the HOD.

5.11 Out of a total of 200 marks for the project work, 50 marks shall be for internal assessment and 150 marks External examination (Viva-voce). A minimum of 50% of maximum marks shall be obtained to earn the corresponding credits.

5.12 If he/she fails to secure those marks he/she will retake the viva-voce examination after three months. If he/she fails to secure those marks at this second viva-voce examination, he will not be eligible for the award of the degree unless the candidate is asked to revise and resubmit. If he/she fails to secure those marks again, the project shall be summarily rejected. Head of the Department shall coordinate and make arrangements for the conduct of Viva- Voce examination.

6. Cumulative Grade Point Average (CGPA)

Marks Range(Max – 100)	Level	Letter Grade	Grade Point
≥ 90	Outstanding	A+	10
≥80 to <90	Excellent	A	9
≥70 to <80	Very Good	B	8
≥60 to <70	Good	C	7
≥50 to <60	Satisfactory	D	6
<50	Fail	F	0
-	Absent	AB	0

Computation of SGPA

The following procedure is to be adopted to compute the Semester Grade Point Average (SGPA) and Cumulative Grade Point Average (CGPA):

The **SGPA** is the ratio of sum of the product of the number of credits with the grade points scored by a student in all the courses taken by a student and the sum of the number of credits of all the courses undergone by a student, i.e.,

$$\textbf{SGPA (Si)} = \sum \textbf{(Ci X Gi)} / \sum \textbf{Ci}$$

Where C_i is the number of credits of the i^{th} course and G_i is the grade point scored by the student in the i^{th} course.

Computation of CGPA

The **CGPA** is also calculated in the same manner taking into account all the courses undergone by a student over all the semester of a programme, i.e.,

$$\textbf{CGPA} = \sum \textbf{(Ci X Si)} / \sum \textbf{Ci}$$

Where Si is the SGPA of the i^{t} semester and C_i is the total number of credits in that semester. The SGPA and CGPA shall be rounded off to TWO decimal points and reported in the transcripts.

7. AWARD OF DEGREE AND CLASS

After a student has satisfied the requirements prescribed for the completion of the program and is eligible for the award of M.C.A. Degree he shall be placed in one of the following four classes:

Class Awarded	CGPA to be secured	
First Class with Distinction	≥ 7.75 (Without any supplementary appearance)	**From the CGPA secured from 80 Credits.**
First Class	≥ 7.75 (With any supplementary appearance) ≥ 6.75 to < 7.75	
Second Class	≥ 6.0 to < 6.75	
Pass Class	≥ 5.0 to < 6.0	

The secured grade, grade points, status and credits obtained will be shown separately in the memorandum of marks.

8. WITHHOLDING OF RESULTS

If the student is involved in indiscipline/malpractices/court cases, the result of the student will be withheld.

9. TRANSITORY REGULATIONS

9.1 Discontinued or detained candidates are eligible for readmission (within the duration as mentioned in item 2.1) as and when next offered. \

9.2 The readmitted students will be governed by the regulations under which the candidate has been admitted.

10. MINIMUM INSTRUCTION DAYS

The minimum instruction days for each semester shall be 90 working days. There shall be no transfer from one college to another within the Constituent Colleges and Units of Jawaharlal Nehru Technological University Kakinada.

11. GENERAL

11.1 Wherever the words “he”, “him”, “his”, occur in the regulations, they include “she”, “her”, “hers”.

11.2 The academic regulation should be read as a whole for the purpose of any interpretation.

11.3 In the case of any doubt or ambiguity in the interpretation of the above rules/ regulations, the decision of the Vice-Chancellor is final.

11.4 The Institute may change or amend the academic regulations or syllabi at any time and the changes or amendments made shall be applicable to all the students with effect from the dates notified by the University.

***

MALPRACTICES RULES DISCIPLINARY ACTION FOR / IMPROPER CONDUCT IN EXAMINATIONS

	Nature of Malpractices/Improper conduct	**Punishment**
	If the candidate:	
1. (a)	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)	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.	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.
3.	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 External 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.	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 External examinations. The continuation of the course by the candidate is subject to the academic regulations in connection with forfeiture of seat.
5.	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.	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.	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 External examinations. The continuation of the course by the candidate is subject to the

		academic regulations in connection with forfeiture of seat.
8.		Expulsion from the examination hall and cancellation of the performance in that subject
	Possess any lethal weapon or firearm in the examination hall.	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 colleges 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.	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.	

COURSE STRUCTURE

I SEMESTER

S.No	Course Code	Course Name	Category	L	T	P	C	IM	EM	TM
1	24MC1T01	Data Structures	PC	3	0	0	3	40	60	100
2	24MC1T02	Computer Organization	PC	3	0	0	3	40	60	100
3	24MC1T03	Database Management Systems	PC	3	0	0	3	40	60	100
4	24MC1T04	Operating Systems	PC	3	0	0	3	40	60	100
5	24MC1T05	Mathematical and Statistical Foundations	BS&H	3	1	0	4	40	60	100
6	24MC1L01	Database Management Systems Lab	PC	0	0	3	1.5	40	60	100
7	24MC1L02	Data Structures using C Lab	PC	0	0	4	2	40	60	100
8	24MC1L03	Operating Systems and Linux Lab	PC	0	0	3	1.5	40	60	100
Total				**15**	**1**	**10**	**21**	**320**	**480**	**800**

II SEMESTER

S.No	Course Code	Course Name	Category	L	T	P	C	IM	EM	TM
1	24MC2T01	Computer Networks	PC	3	0	0	3	40	60	100
2	24MC2T02	Network Security and Cyber Security	PC	3	0	0	3	40	60	100
3	24MC2T03	Object Oriented Programming Using JAVA	PC	3	0	0	3	40	60	100
4	24MC2T04	Software Engineering	PC	3	0	0	3	40	60	100
5	24MC2T05	Artificial Intelligence	PC	3	0	0	3	40	60	100
6	-	Program Elective-1 1. Design and Analysis of Algorithms 2. Advanced Unix Programming 3. Data Warehousing and Data mining 4. MOOCS-1(NPTEL/SWAYAM) (Recommended 12 week course with 3 credits)	PC/ PE	3	0	0	3	40	60	100
7	24MC2L01	Object Oriented Programming Using JAVA Lab	PC	0	0	3	1.5	40	60	100
8	24MC2L02	Networks and Security Lab	PC	0	0	3	1.5	40	60	100
9	24MC2L03	Employability Skills-1$	AC	1	0	0	0	-	-	-
Total				**19**	**0**	**6**	**21**	**320**	**480**	**800**

III SEMESTER
Tentative Course Structure for III and IV Semester

S.No	Course Code	Course Name	Category	L	T	P	C	IM	EM	TM
1	24MC3T01	Theory	PC		0	0	3	40	60	100
2	24MC3T02	Theory	PC	3	0	0	3	40	60	100
3	24MC3T03	Theory	PC	3	0	0	3	40	60	100
4	24MC3T04	Theory	PC	3	0	0	3	40	60	100
5		Theory Elective	PE	3	0	0	3	40	60	100
6		Theory Elective	PE	3	0	0	3	40	60	100
7	24MC3L01	Lab	PC	0	0	3	1.5	40	60	100
8	24MC3L02	Lab	PC	0	0	3	1.5	40	60	100
9	24MC3L03	Internship / industry oriented mini project/ skill development course	PC	0	0	0	1	50	-	50
Total				**18**	**0**	**6**	**22**	**370**	**480**	**850**

IV SEMESTER

S.No	Course Code	Course Name	Category	L	T	P	C	IM	EM	TM
1	24MCP01	Project Work	AC	1	0	0	16	40	60	100
Total								40	60	100

Program Elective-1

S. NO	Course Code	Course Title
1	24MC2TE1	Design and Analysis of Algorithms
2	24MC2TE2	Advanced Unix Programming
3	24MC2TE3	Data Warehousing and Data mining
4	24MC2TE4	MOOCS-1(NPTEL/SWAYAM) (Recommended 12 week course with 3 credits)

I Semester	L	T	P	C
	3	0	0	3
24MC1T01 :: DATA STRUCTURES				

Course Objectives:

- Explore basic data structures such as stacks and queues.
- Introduce various data structures such as hash tables, search trees, tries, heaps, graphs, sorting, and pattern matching algorithms.

Course Outcomes:

Course Outcomes		Knowledge Level (K)#
CO1	Implement basic programs by using C concepts.	**K2**
CO2	Implement C Program using Functions, Structures and Unions, Pointers	**K3**
CO3	Design advanced Data Structures using Non Linear Data Structures	**K4**
CO4	Create Hash Table for storing data	**K4**
CO5	Apply appropriate Sorting technique for a problem	**K5**

UNIT-I:

Introduction to C: Constants and variables, Operators and Expressions, Managing Input and Output operators, Decision making - branching and looping, Arrays.

UNIT-II:

Functions, Structures and Unions, Pointers, File handling in C.

UNIT-III:

Data structure: Definition, types of data structures. Recursion Definition, Design Methodology and Implementation of recursive algorithms, Linear and binary recursion. Preliminaries of algorithms, analysis, and complexity. **Linear list** – singly linked list, double linked list, and circular linked list - implementation, insertion, deletion, and searching operations on linear list.

UNIT-IV:

Stacks - Operations, array and linked representations of stacks, stack applications. **Queues** - Operations, array and linked representations**. Hash Table Representation**: hash functions, collision resolution - separate chaining, open addressing - linear probing, quadratic probing, double hashing, and rehashing, extendible hashing.

UNIT-V:

Sorting Techniques: Insertion sort, selection sort, exchange-bubble sort, quick sort, and merge sort algorithms. **Trees:** Binary Trees, terminology, representation, and traversals - pre, post, in order traversals. **Search Trees:** Binary Search Trees, Definition, Implementation, Operations - Searching, Insertion, and Deletion, AVL Trees, Red-Black Trees.

Text Books:

1. Data structures using C Reema Thareja, Oxford University Press, 2014
2. Data Structures Through C In Depth, Second Revised & Update Edition Paperback – 1 January 2021, by S.K. Srivastava Deepali Srivastava
3. Programming in ANSI C, 5e, E. Balagurusamy, TMH.
4. Fundamentals of Data Structures in C, 2nd Edition, E. Horowitz, S. Sahni and Susan Anderson Freed, Universities Press.

Reference Books:

1. Data Structures: A Pseudocode Approach with C, 2nd Edition, R.F. Gilberg and B. A. Forouzan, Cengage Learning.

Web Resources:

1. https://archive.nptel.ac.in/courses/106/102/106102064/
2. https://ocw.mit.edu/courses/6-006-introduction-to-algorithms-spring-2020/video_galleries/lecture-videos/
3. https://www.cs.usfca.edu/~galles/visualization/Algorithms.html
4. https://visualgo.net/en
5. https://elearn.daffodilvarsity.edu.bd/course/view.php?id=11771

I Semester	L	T	P	C
	3	0	0	3
24MC1T02 :: COMPUTER ORGANIZATION				

Course Objectives:

- Conceptualize the basics of organizational and architectural issues of a digital computer.
- Learn the function of each element of a memory hierarchy.
- Study various data transfer techniques in digital computers.

Course Outcomes:

Course Outcomes		Knowledge Level (K)#
CO1	Understand the basic types of computers and their key functional units.	**K2**
CO2	Develop and implement programs using machine instructions and addressing modes, including stack and queue operations.	**K3**
CO3	Analyse various input/output techniques like interrupt-driven I/O, Direct Memory Access (DMA), and standard I/O interfaces.	**K4**
CO4	Understand and evaluate the performance trade-offs in memory systems, focusing on RAM, ROM, cache, and virtual memory.	**K5**
CO5	Explain the concepts of parallel processing, including pipeline processors and multiprocessor systems, and their impact on performance.	**K2**

UNIT-I:

Basic Structure Of Computers: Computer Types, Functional units, Basic Operational concepts, Bus structures, Software, Performance, multiprocessor and multi computers, Historical perspective.

UNIT-II:

Machine Instructions and Programs: Numbers, Arithmetic Operations, and Characters, Memory locations and addresses, Memory operations, Instructions and Instruction sequencing, Addressing Modes, Assembly Languages, stacks and Queues Basic Input/output Operations, role of Stacks and Queues Additional Instructions.

UNIT-III:

Input/output Organization: Accessing I/O Devices, Interrupts, Processor examples, Direct Memory Access, Buses, Interface Circuits, and Standard I/O Interfaces.

UNIT-IV:

The Memory Systems: Some Basic concepts, Semiconductor RAM memories, Memory System Consideration, Read-Only Memories, Speed, Size, and cost, Cache Memories, Performance considerations, Virtual Memories, Memory Management Requirements, Secondary Storage.

UNIT-V:

Parallel Processing: Basic concepts, Pipeline Processors, Multiprocessors.

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 of Computer Organization and Design, Siva ramaDandamudi, Springer Int. Edition.

Web Resources:

1. https://nptelvideos.com/course.php?id=396
2. https://onlinecourses.nptel.ac.in/noc20_cs64/preview
3. https://www.learncomputerscienceonline.com/computer-organization-and-architecture/
4. http://williamstallings.com/COA/COA8e-student/index.html

I Semester	L	T	P	C
	3	0	0	3
24MC1T03 :: DATABASE MANAGEMENT SYSTEMS				

Course Objectives:

- Explain the concept of databases, database management systems, database structures, and their functioning.
- Use Entity-Relationship Modelling and Relational Modelling for creating databases from real-world scenarios.
- Write relational algebra and SQL statements.
- Normalize databases using Normalization Rules.
- Discuss Transaction Management, Recovery, Tree Structured and Hash-Based Indexing issues.

Course Outcomes:

Course Outcomes		Knowledge Level (K)#
CO1	Explain the purpose, architecture, and various applications of database systems, as well as the role of data models (ER models) and how they relate to database design.	**K2**
CO2	Demonstrate the application of the relational model, integrity constraints, and query relational data using relational algebra and calculus. Convert ER diagrams into relational schema and construct basic SQL queries.	**K3**
CO3	Analyze and construct SQL queries, including nested and aggregate queries, constraints, and triggers. Evaluate the normalization process using functional dependencies to achieve higher normal forms.	**K4**
CO4	Evaluate schema refinement methods, such as multivalued dependencies and normal forms, and assess transaction management strategies, including concurrency control and recovery protocols in database systems.	**K5**
CO5	Design and implement storage solutions and indexing mechanisms, such as hash-based and tree-based indexing (B+ trees, ISAM), optimizing file organization and performance for database queries.	**K6**

UNIT-I:

Overview of Database System: Database System Applications, Purpose of Database System, View of Data, Database Languages, Relational Databases, Database Design, Data Storage and Querying, Transaction Management, Database Architecture, Data Mining and Information Retrieval, Specialty Databases, Database Users and Administrators, History of Database Systems. [Text Book – 2] **Introduction to Database Design:** Database Design and ER Diagrams, Entities, Attributes and Entity Sets, Relationships and Relationship Sets, Additional Features of ER Model, Conceptual Design with the ER Model, Extended ER features [Text Book =1]

UNIT-II:

Relational Model: Introduction to the Relational Model, Integrity Constraints over Relations, Enforcing Integrity Constraints, Querying Relational Data, Logical Database Design: ER to Relational, Introduction to Views, Destroying/Altering Tables and Views [Text Book =1] **Relational Algebra:** Selection and Projection, Set Operations, Renaming, Joins, Division, More Examples of Algebra Queries, **Relational Calculus:**

Tuple Relational Calculus, Domain Relational Calculus [Text Book -1]

UNIT-III:

SQL: Queries, Constraints, Triggers: The Form of a Basic SQL Query, UNION, INTERSECT and EXCEPT, Nested Queries, Aggregate Operators, Null Values, Complex Integrity Constraints in SQL, Triggers, Exceptions, Procedures, Functions [Text Book -1] **Normal Forms:** Introduction to Schema Refinement, Functional Dependencies, Reasoning about FDs, Normal Forms, Properties of Decompositions, Normalization. [Text Book -1]

UNIT-IV:

Schema Refinement: Multivalued dependencies, Fourth Normal form, Join Dependencies, Fifth Normal Form, Lossless join, dependency preservation. [Text Book -1] **Transaction Management:** Transaction Concepts, Transaction State, Implementation of Atomicity and Durability, Concurrent Execution, Serializability, Recoverability. [Text Book -2] **Concurrency Control:** Lock-based Protocols: Locks, Granting of Locks, Two Phase Locking Protocol, Implementation of locking; Timestamp-Based Protocols: Time Stamps, Time Stamp Ordering protocol, Thomas Write Rule, Validation-Based Protocols. [Text Book -2]

UNIT-V:

Overview of Storage and Indexing: Data on External Storage, File organization and indexing: Clustered Indexes, Primary and Secondary Indexes; Index Data Structures: Hash and Tree based indexing; Comparison of File organizations. [Text Book -1] **Tree Structured Indexing:** Intuitions for Tree Indexes, Indexed Sequential Access Method (ISAM), B+ Trees: A Dynamic Index Structure, Search, Insert, Delete, Duplicates, B+ Trees in Practice [Text Book -1]

Text Books:

1. Database Management Systems, 3/e, Raghurama Krishnan, Johannes Gehrke, McGraw-Hill.
2. Database System Concepts, 6/e, Abraham Silberschatz, Henry F. Korth, S. Sudarshan, McGraw-Hill.
3. Database Systems, 9/e, Carlos Coronel, Steven Morris, Peter Rob, Cengage.

Reference Books:

1. Introduction to Database Systems, 8/e, CJ Date, Pearson.
2. Database Systems, 6/e RamezElmasri, Shamkant B. Navathe, Pearson.

Web Resources:

1. https://nptel.ac.in/courses/106105175
2. https://onlinecourses.swayam2.ac.in/cec22_cs18/preview
3. https://cs186berkeley.net/
4. https://www.youtube.com/playlist?list=PL52484DF04A264E59
5. https://courses.cs.washington.edu/courses/cse414/17au/calendar/lecturelist.html
6. https://www.db-book.com/slides-dir/index.html

I Semester	L	T	P	C
	3	0	0	3
24MC1T04 :: OPERATING SYSTEMS				

Course Objectives:

- Introduce different types of operating systems.
- Learn process management techniques.
- Learn various memory management techniques.
- Introduce the architecture of the Linux operating system.
- Learn about Unix and Windows operating systems.

Course Outcomes:

Course Outcomes		Knowledge Level (K)#
CO1	Explain the fundamental concepts and structure of operating systems, including types and system calls.	**K2**
CO2	Apply process scheduling algorithms and analyze process management concepts such as process control blocks and inter process communication.	**K3**
CO3	Analyze synchronization mechanisms and deadlock handling strategies to ensure safe process execution.	**K4**
CO4	Evaluate memory management techniques, such as paging and segmentation, and file system implementation strategies like allocation and disk scheduling.	**K5**
CO5	Design and compare the operating system architectures of Linux and Windows, focusing on their process management, file systems, and networking features.	**K6**

UNIT-I:

Introduction to Operating System Concept: Types of Operating Systems, Operating Systems Concepts, Operating System Operations. Operating Systems Structures Operating System Services, User Operating-System Interface, Introduction to System calls, Types of System Calls.

UNIT-II:

Process Management: Process concept, Process State Diagram, Process Control Block, Process Scheduling, Inter process Communication, Threads Threading Issues, Scheduling - Basic Concepts, Scheduling Criteria, Scheduling Algorithms.

UNIT-III:

Process Synchronization: Critical-Section Problem, Peterson's Solution, Synchronization Hardware, Semaphores, Classic Problems of Synchronization, Monitors, **Principles of deadlock:** System Model, Deadlock characterization, Deadlock handling, Deadlock Prevention, Detection and Avoidance, Recovery Starvation, Critical Regions form Deadlock

UNIT-IV:

Memory Management: Swapping, Contiguous Memory Allocation, Paging, structure of the Page Table, Segmentation Virtual Memory Management- Demand Paging, Page Replacement Algorithms, Thrashing. **File-**

System Interface: File Concept, Access Methods, Directory structure, File- System mounting, Files Sharing, Protection. File- System implementation- File- System Structure, Allocation Methods, Free- Space Management, Disk Structure, Disk Scheduling

UNIT-V:

Case Studies: Linux System: Design Principles, kernel Modules, Process Management, File Systems, Input and Output, Inter process Communication, Network Structure, Security. **Windows 7:** Design Principles, System Components, Terminal Services and Fast User, File System, Networking, Programmer Interface.

Text Books:

1. Operating System Concepts, Abraham Siliberschatz, Peter Baer Galvin, Greg Gagne, John Wiley & Sons, Inc., Edition 9,2011
2. Introduction to UNIX and Shell Programming, M.G.VenkateshMurhty, Pearson, 2005
3. UNIX & Shell Programming, B.M.Harwani, OXFORD University Press,2013

Reference Books:

1. Advanced Programming in the UNIX Environment, W.Richard Stevens, Stephen Rago, Wesley Professional, 2013
2. UNIX Network Programming, W.Richard Stevens,1990
3. Operating Systems, William stallings, PHI/Pearson,6/E,2009
4. Operating Systems, Dietal, Pearson,3/e,2007
5. Operating Systems, Dhamdhere,TMH,2/e,2009

Web References:

1. https://onlinecourses.swayam2.ac.in/cec20_cs06/preview
2. https://www.cse.iitb.ac.in/~mythili/os/
3. https://onlinecourses.nptel.ac.in/noc21_cs72/preview
4. https://web.stanford.edu/~ouster/cgi-bin/cs140-spring20/lectures.php
5. https://oscourse.org/
6. https://www.cs.jhu.edu/~huang/cs318/fall21/schedule.html

I Semester	L	T	P	C
	3	1	0	4
24MC1T05 :: MATHEMATICAL AND STATISTICAL FOUNDATIONS				

Course Objectives:

This course is aimed at enabling the students to

- To understand the mathematical fundamentals that are prerequisites for a variety of courses such as Data Mining, Network Protocols, Analysis of Web Traffic, Computer Security, Software Engineering, Computer Architecture, Operating Systems, Distributed Systems, Bioinformatics, and Machine Learning.
- To develop an understanding of the mathematical and logical techniques used in modern computer science technologies, including Machine Learning, Programming Language Design, and Concurrency.
- To study various sampling and classification problems.

Course Outcomes:

Course Outcomes		Knowledge Level (K)#
CO1	Understand the basic concepts of probability, random variables, and probability distributions for discrete and continuous variables.	**K2**
CO2	Apply sampling methods and estimation techniques to compute population parameters and evaluate point and interval estimates.	**K3**
CO3	Analyze and conduct hypothesis tests, including significance tests for small and large samples, and apply chi-square tests for goodness of fit.	**K4**
CO4	Evaluate algebraic structures such as groups, monoids, and homomorphisms, and apply number theory concepts like Euclidean algorithms and modular arithmetic.	**K3**
CO5	Design and analyze graphs using concepts like Eulerian and Hamiltonian circuits, graph coloring, and spanning trees, applying algorithms for practical problems.	**K4**

UNIT I:

Basic Probability and Random Variables: Random Experiments, Sample Spaces Events, the Concept of Probability ,The Axioms of Probability, Some Important Theorems on Probability, Assignment of Probabilities, Conditional Probability Theorems on Conditional Probability Independent Events, Bayes Theorem or Rule Random Variables, Discrete Probability Distributions, Distribution Functions for Random, Variables Distribution Functions for Discrete Random, Variables Continuous Random Variables

UNIT II:

Sampling and Estimation Theory: Population and Sample, Statistical Inference Sampling With and Without Replacement Random Samples, Random Numbers Population Parameters, Sample Statistics Sampling Distributions, Frequency Distributions Relative Frequency Distributions, Computation of Mean, Variance, and Moments for Grouped Data. Unbiased Estimates and Efficient Estimates Point Estimates and Interval Estimates. Reliability Confidence Interval Estimates of Population Parameters, Maximum Likelihood Estimates

UNIT III:

Tests of Hypothesis and Significance: Statistical Decisions, Statistical Hypotheses, Null Hypotheses Tests of

Hypotheses and Significance Type I and Type II Errors Level of Significance Tests Involving the Normal Distribution One-Tailed and Two-Tailed Tests P Value Special Tests of Significance for Large Samples Special Tests of Significance for Small Samples Relationship between Estimation Theory and Hypothesis Testing Operating Characteristic Curves. Power of a Test Quality Control Charts Fitting Theoretical Distributions to Sample Frequency Distributions. The Chi-Square Test for Goodness of Fit Contingency Tables Yates' Correction for Continuity Coefficient of Contingency.

UNIT IV:

Algebraic Structures and Number Theory: Algebraic Systems, Examples, General Properties, Semi-Groups and Monoids, Homomorphism of Semi Groups Semi Group and Monoids, Group Subgroup, Abelian Group, Homomorphism, Isomorphism, Properties of Integers, Division Theorem, The Greatest Common Divisor, Euclidean Algorithm, Least Common Multiple, Testing for Prime Numbers, The Fundamental Theorem of Arithmetic, Modular Arithmetic (Fermat's Theorem and Euler's Theorem)

UNIT V:

Graph Theory: Basic Concepts of Graphs, Sub graphs, Matrix Representation of Graphs: Adjacency Matrices, Incidence Matrices, Isomorphic Graphs, Paths and Circuits, Eulerian and Hamiltonian Graphs, Multigraphs, Planar Graphs, Euler's Formula, Graph Coloring and Covering, Chromatic Number, Spanning Trees, Algorithms and Spanning Trees (Problems Only and Theorems without Proofs)

Text Books:

1. Foundation Mathematics for Computer Science, 1st Edition, John Vince, Springer, 2015.
2. Probability & Statistics, 3rd Edition, Murray R. Spiegel, John J. Schiller, R. Alu Srinivasan, Schaum's Outline Series, Tata McGraw-Hill Publishers, 2018.
3. Probability and Statistics with Reliability, 2nd Edition, K. Trivedi, Wiley, 2011.
4. Discrete Mathematics and its Applications with Combinatorics and Graph Theory, 7th Edition, H. Rosen, Tata McGraw Hill, 2003.

Reference Books:

1. Probability and Computing: Randomized Algorithms and Probabilistic Analysis, 1st Edition, M. Mitzenmacher and E. Upfal, 2005.
2. Applied Combinatorics, 6th Edition, Alan Tucker, Wiley, 2012.

Web Resources:

1. https://archive.nptel.ac.in/courses/106/102/106102064/
2. https://ocw.mit.edu/courses/6-006-introduction-to-algorithms-spring-2020/video_galleries/lecture-videos/
3. https://www.cs.usfca.edu/~galles/visualization/Algorithms.html
4. https://visualgo.net/en
5. https://elearn.daffodilvarsity.edu.bd/course/view.php?id=11771

I Semester	L	T	P	C
	0	0	3	1.5
24MC1L01 :: DATABASE MANAGEMENT SYSTEMS LAB				

Course Objectives:
This Course will enable students to

- Populate and query a database using SQLDDL/DML Commands.
- Declare and enforce integrity constraints on a database
- Writing Queries using advanced concepts of SQL
- Programming PL/SQL including procedures, functions, cursors and triggers

Course Outcomes		Knowledge Level (K)#
CO1	Explain the purpose, architecture, and various applications of database systems, as well as the role of data models (ER models) and how they relate to database design.	**K2**
CO2	Demonstrate the application of the relational model, integrity constraints, and query relational data using relational algebra and calculus. Convert ER diagrams into relational schema and construct basic SQL queries.	**K3**
CO3	Analyze and construct SQL queries, including nested and aggregate queries, constraints, and triggers. Evaluate the normalization process using functional dependencies to achieve higher normal forms.	**K4**
CO4	Evaluate schema refinement methods, such as multivalued dependencies and normal forms, and assess transaction management strategies, including concurrency control and recovery protocols in database systems.	**K5**
CO5	Design and implement storage solutions and indexing mechanisms, such as hash-based and tree-based indexing (B+ trees, ISAM), optimizing file organization and performance for database queries.	**K6**

1. Execute all DDL, DML and DCL commands on sample tables.
2. Implementation of different types of operators and built-in functions with Suitable examples.
3. Implementation of different types of joins with suitable examples.
4. Create views, partitions, Sequences for a particular DB.
5. Implement different types of constraints on relations.
6. Implementation of subqueries and nested queries.
7. Implement Queries on Group By & Having Clauses, ALIAS, Sequence By, Order By.
8. Control Structure
   a) Write a PL/SQL block for Addition of Two Numbers.
   b) Write a PL/SQL block for IF, IF and else condition.
   c) Write a PL/SQL block for implementation of loops.

d) Write a PL/SQL block for greatest of three numbers using IF AND ELSE IF.

9. Exception Handling-Implement the following with respect to exception Handling. Raising Exceptions, User Defined Exceptions, Pre-Defined Exceptions.
10. Procedures:
    a) Write a PL/SQL Procedure using Positional Parameters.
    b) Write a PL/SQL Procedure using notational parameters.
    c) Write a PL/SQL Procedure for GCD Numbers.
    d) Write a PL/SQL Procedures for cursor implementation (explicit and implicit cursors)
11. Functions:
    a) Write a PL/SQL block to implement factorial using functions.
    b) Write a PL/SQL function to search an address from the given database.
12. Write a DBMS program to prepare Pl/SQL reports for an application using functions.
13. Triggers:
    a) Write a Trigger to pop-up the DML operations.
    b) Write a Trigger to check the age valid or not Using Message Alert.
    c) Create a Trigger on a table so that it will update another table while inserting values.
14. Write PL/SQL block for an application using cursors and all types of triggers.
15. Write a PL/SQL block for transaction operations of a typical application using package.

I Semester	L	T	P	C
	0	0	4	2
24MC1L02 :: DATA STRUCTURES LAB				

Course Objectives:

This Course will enable students to

- Design and implement various data structures.
- Implement operations like searching, insertion, and deletion, traversing mechanism.
- Develop applications using data structure algorithms.

Course Outcomes		Knowledge Level (K)#
CO1	Apply basic C programming concepts to solve mathematical problems like even numbers, harmonic series, Armstrong numbers, and factorials.	**K3**
CO2	Apply C programming to perform matrix operations and recursion-based tasks such as Fibonacci sequence generation and call-by-reference operations.	**K3**
CO3	Analyze and implement file handling operations, recursive algorithms, and searching techniques like linear and binary search using both recursive and non-recursive approaches.	**K4**
CO4	Design and implement data structures such as stacks, queues, and linked lists using arrays and linked list techniques in C.	**K6**
CO5	Create and implement advanced data structures such as binary search trees, AVL trees, and hash tables, and apply sorting algorithms like quicksort, merge sort, and bubble sort in C.	**K6**

Experiment 1:

a) Write a program in C to display the n terms of even natural numbers and their sum.
b) Write a program in C to display the n terms of harmonic series and their sum.1+1/2+1/3+1/4+1/5….1/n terms.
c) Write a C program to check whether a given number is an Amstrong number or not.
d) Write a C program to calculate the factorial of given number.

Experiment 2:

a) Write a program in C for multiplication of two square Matrices.
b) Write a program in C to find the transpose of a given matrix.

Experiment 3:

a) Write a program in C to check whether a number is a primer number or not using the function.
b) Write a recursive program which computes the nth Fibonacci number, for Appropriate values of n.
c) Write a program in C to add numbers using call by reference.

Experiment 4:

a) Write a program in C to append multiple lines at the end of a text file.
b) Write a program in C to copy a file in another name.

Experiment 5:

Write recursive program for the following:

a) Write recursive and non- recursive C program for calculation of Factorial of an integer.
b) Write recursive and non -recursive C program for calculate of GCD(n, m)
c) Write recursive and non-recursive C program for Towers of Hanoi: N disks are to be transferred from peg s to peg D with peg I as the intermediate peg.

Experiment 6:

a) Write a C program that uses both recursive and non -recursive functions to Perform Linear search for a key value in a given list.
b) Write a C program that uses both recursive and non-recursive function to perform Binary search for key value in a given list.

Experiment 7:

a) Write a C program that implements stack (its operations) using arrays.
b) Write a C program that implements stack (its operations) using Linked list.

Experiment 8:

a) Write a C program that uses Stack operations to convert infix expressions into postfix expressions.
b) Write a C program that implements Queue (its operations) using arrays.
c) Write a C program that implements Queue (its operations) using linked lists.

Experiment 9:

- Write a C program that uses functions to create a single linked list and perform various operations.

Experiment 10:

- Write a C program to store a polynomial expression in memory using a linked list and perform polynomial addition.

Experiment 11:

a) Write a recursive C program for traversing a binary tree in preorder, in order and post order.
b) Write a non-recursive C program for traversing a binary tree in preorder, in order and post order.

Experiment 12:

- Implementation of Hash table using double hashing as collision resolution Function.

Experiment 13:

- Implementation of Binary Trees-Insertion and deletion.

Experiment 14:

- Implementation of AVL Tree – Insertion and deletion.

Experiment 15:

a) Write a c program that implements Bubble sort, to sort a given list of integers in ascending order.
b) Write a C program that implements Quick sort a given list of integers in ascending order.
c) Write a C program that implements Merge sort, to sort a given list of integers in ascending order.

I Semester	L	T	P	C
	0	0	3	1.5
24MC1L03 :: OPERATING SYSTEMS AND LINUX LAB				

Course Objectives:

- This Course will enable students to implement CPU scheduling
- Algorithms, disk scheduling algorithms, Execute different types of
- Linux commands and write shell scripts.

Course Outcomes:

Course Outcomes		Knowledge Level (K)#
CO1	Understand and use basic Unix/Linux utility commands and shell environments like Bash, Bourne, and C shell for effective system interaction.	**K2**
CO2	Apply C programming to simulate Unix/Linux file system operations and process control functions, including command pipes and system calls like fork (), wait (), and exec ().	**K3**
CO3	Apply and simulate various CPU scheduling algorithms (FCFS, SJF, Priority, Round Robin) and memory management strategies like first-fit, best-fit, and worst-fit in operating systems.	**K3**
CO4	Analyze and simulate deadlock avoidance/prevention techniques using the Banker's Algorithm and implement page replacement algorithms like FIFO, LRU, and LFU.	**K4**
CO5	Design and develop shell scripts for various tasks, including checking prime numbers, calculating Fibonacci series, handling file operations, and manipulating student records with conditions and loops.	**K6**

List of Experiments:

UNIX Lab- Introductions to Unix

1. Study of Unix/Linux general purpose utility commands.
2. Study of bash shell, Bourne shell and C shell in Unix/Linux operating system.
3. Study of UNIX/LINUX file system (tree structure).
4. C program to emulate the Unix ls – 1 command
5. C program that illustrates how to execute two commands concurrently with a command pipe. Ex: -1 sort.
6. Multiprogramming memory management- Implementation of fork (), wait (), exec() and exit (), System calls

Operating System Lab

1. Simulating the Following CPU Scheduling Algorithms
   A) FCFS B) SJF C) priority D) Round Robin
2. Multiprogramming-Memory Management-Implementation of fork (), wait (), exec () and exit ()
3. Simulating The Following
   a. Multiprogramming with A Fixed Number of Task (MFT)

b. Multiprogramming with A Variable Number of Task (MVT)

4. Write program to implement first fit, best and worst fit algorithm for memory management.
5. Simulate Bankers Algorithm for Deadlock Avoidance.
6. Simulate Bankers Algorithm for Deadlock prevention.
7. Simulate the Following page replacement Algorithms.
   a) FIFO b) LRU c) LFU
8. Simulate the Following File Allocation strategies
   a) Sequenced b) Indexed c) Linked

Linux Lab

1. Write a shell program to check whether a given number is prime or not.
2. Write a shell script which will display Fibonacci series up to the given range.
3. Write a shell script to check whether the given number is Amstrong or not.
4. Write a shell script to the calculate the value of
5. Write a shell script to accept student number, name, marks in 5 subjects.
6. Find total average and grade using the following rules:

   Avg>=80 then grade A

   Avg<80 &&Average >=70 then grade B

   Avg<70 &&Average >=60 then grade C

   Avg<60 &&Average >=50 then grade D

   Avg<50 &&Average >=40 then grade E
7. Write a shell script to find minimum and maximum elements in the

given list of the elements.

8. Write shell program to check whether the given string is palindrome or not.
9. Write an AWK program to print sum, average of students marks list.

Write a shell script to compute number of characters and words in each line of given file.

II Semester	L	T	P	C
	3	0	0	3
24MC2T01 :: COMPUTER NETWORKS				

Course Objectives:

By the end of this course, students will be able to:

1. Understand fundamental concepts of computer networking and the OSI Reference Model.
2. Familiarize themselves with the basic taxonomy and terminology of computer networking.
3. Learn and grasp advanced networking concepts to prepare for advanced courses in computer networking.
4. Develop expertise in specific areas of networking such as the design and maintenance of individual networks.

Course Outcomes		Knowledge Level (K)#
CO1	Describe network topologies, reference models, and physical layer components, including different media types.	**K2**
CO2	Apply error detection/correction and implement basic data link protocols for reliable communication.	**K3**
CO3	Compare multiple access protocols and evaluate their effectiveness in various networking environments.	**K4**
CO4	Analyze routing algorithms and congestion control techniques to optimize data transmission.	**K4**
CO5	Evaluate transport layer protocols and application services for end-to-end communication and data security.	**K5**

UNIT I:

Introduction: Network Topologies WAN, LAN, MAN, Reference Models. The OSI Reference Model, the TCP/IP Reference Model, A Comparison of OSI and TCP/IP Models, **Physical Layer**: Introduction to physical layer, Data and Signals, Periodic analog Signals, Digital Signals, Transmission Impairment, Data Rate Limits, Performance, Introduction to Guided Media Twisted-pair cable, Coaxial cable, Fiber optic cable, Unguided Media : Wireless - Radio waves, Microwaves, Infrared

UNIT II:

The Data Link Layer: Services Provided to the Network Layer**,** Framing, Error Control, Flow Control, Error Detection and Correction, Error-Correcting Codes, Error-Detecting Codes, **Elementary Data Link Protocols**: Simplex Protocol, A Simplex Stop-and-Wait Protocol for an Error free channel, A Simplex Stop and Wait Protocol for a Noisy Channels, Sliding Window Protocols, A One Bit Sliding Window Protocol, Go-Back-N, Selective Repeat

UNIT III:

The Medium Access Control Sublayer : The Channel Allocation Problem, Static Channel Allocation, Assumptions for Dynamic Channel Allocation. Multiple Access Protocols, Aloha, Pure Aloha, Slotted Aloha, Carrier Sense Multiple Access Protocols, Collision-Free Protocols, Limited Contention Protocols. **Wireless LAN Protocols**-Ethernet, Classic Ethernet Physical Layer, Classic Ethernet MAC Sublayer Protocol, Ethernet Performance, Fast Ethernet, Wireless LANs, The 802.11 Architecture and Protocol Stack, 802.11 Physical Layer, 802.11 MAC Sublayer Protocol, 805.11 Frame Structure, Services

UNIT IV:

The Network Layer Design Issues: Store-and-Forward Packet Switching, Services Provided to Transport Layer, Implementation of Connectionless Services, Implementation of Connection-Oriented Services. Comparison of Virtual Circuit and Datagram Networks, Routing Algorithms, Optimality Principle, Shortest Path, Flooding, Distance Vector, Link State, Hierarchical. **Congestion Control Algorithms**: General Principles of congestion control, Congestion Prevention Policies, Approaches to Congestion Control, Traffic Aware Routing, Admission Control, Traffic Throttling, Load Shedding. **Internetworking**: How networks differ, How networks can be connected, Tunneling, Internetwork routing, Fragmentation network layer in the internet, IP protocols, IPV4 protocol, IP addresses, Subnets, IP Version6- The main IPV6 header, Internet control protocols- ICMP, ARP, DHCP.

UNIT V:

Transport Layer: Transport layer Protocols Introduction, Services, Port Numbers, User Datagram Protocol, User datagram, UDP Services, UDP applications, Transmission Control Protocol: TCP Services- TCP Features- Segments-A TCP Connections, Windows in TCP, Flow Control, Error Control. **Application Layer**: World Wide Web: HTTP, FTP - Two Connections, Control Connection, Data Connection, Security of FTP, Electronic Mail, Architecture, Web-based Mail, email Security, TELNET, Local versus. remote Logging, **Domain Name System**: Name Space, DNS in Internet, Resolution, Caching, Resource Records, DNS Messages, Registrars, Security of DNS Name Servers.

Textbooks:

1. Computer Networks: Andrew S Tanenbaum David J. Wetherall, 5/e, Pearson
2. Data Communications and Networking, BehrouzForouzan, 5/e, McGraw Hill

Reference Books:

1. Computer Networks- A Systems Approach, Peterson, Bruce Davie,2/e, Harcourt Asia
2. Computer Communications and Networking Technologies, Gallo, Hancock, Cengage
3. An Engineering Approach to Computer Networking, Kesha, Pearson

Web Resources:

1. https://onlinecourses.swayam2.ac.in/cec23_cs07/preview
2. https://onlinecourses.nptel.ac.in/noc21_cs18/preview
3. https://ocw.mit.edu/courses/6-829-computer-networks-fall-2002/pages/lecture-notes/
4. https://www.sanfoundry.com/computer-network-basics/
5. https://www.cisco.com/c/en_in/solutions/enterprise-networks/what-is-computer-networking.html
6. https://www.cs.vu.nl/~ast/CN5/

II Semester	L	T	P	C
	3	0	0	3
24MC2T02 :: NETWORK SECURITY AND CYBER SECURITY				

Course Objectives:

1. To learn various cryptographic algorithms including secret key cryptography, hashes and message digests, and public key algorithms.
2. To become familiar with the design issues and working principles of various authentication protocols and secure communication standards.
3. To understand cybercrime fundamentals and preventive measures.

Course Outcomes		Knowledge Level (K)#
CO1	Explain basic cryptography principles, including security goals, attacks, and symmetric encryption techniques like DES and AES.	**K2**
CO2	Apply asymmetric encryption methods and compare cryptographic hash functions such as SHA and SHA-3.	**K3**
CO3	Analyze digital signature schemes and evaluate security measures for email and IP security.	**K4**
CO4	Identify and classify cybercrimes and understand the roles and motivations of cybercriminals.	**K1**
CO5	Evaluate advanced cyber threats and propose security measures to counter them.	**K5**

UNIT I:

Basic Principles: Security Goals, Cryptographic Attacks, Services and Mechanisms, Mathematics of Cryptography. Symmetric Encryption**:** Mathematics of Symmetric Key Cryptography, Introduction to Modern Symmetric Key Ciphers, Data Encryption Standard, Advanced Encryption Standard.

UNIT II:

Asymmetric Encryption: Mathematics of Asymmetric Key Cryptography -Primes, Primality Testing, Factorization. Asymmetric Key Cryptography-RSA Cryptosystem, Rabin Cryptosystem, ElGamal Cryptosystem, Elliptic Curve Cryptosystem. **Cryptographic Hash Functions:** Applications of Cryptographic Hash Functions, Two Simple Hash Functions, Requirements and Security, Hash Functions Based on Cipher Block Chaining, Secure Hash Algorithm (SHA), SHA-3.

UNIT III:

Digital Signatures : ElGamal Digital Signature Scheme, Schnorr Digital Signature, NIST Digital Signature Algorithm.

Electronic Mail Security: Internet Mail Architecture, Email Formats, Email Threats and Comprehensive Email Security, S/MIME. IP Security**:** IP Security Policy, Encapsulating Security Payload, Combining Security Associations, Internet Key Exchange.

UNIT IV:

Introduction to Cybercrime: Introduction, Cybercrime: Definition and Origins of the Word, Cybercrime and Information Security, Cybercriminals, Classifications of Cybercrime, Cyber Stalking, Cyber Café and Cybercrimes, Botnets. Attack Vector, Proliferation of Mobile and Wireless Devices, Security Challenges Posed by Mobile Devices, Attacks on Mobile/Cell Phones.

UNIT V:

Proxy Servers and Anonymizers, Phishing, Password Cracking, Keyloggers and Spywares, Viruses and Worms, Trojan Horses and Backdoors, Steganography, Sniffers, Spoofing, Session Hijacking. Buffer Overflow, DoS and DDoS Attacks, SQL Injection, Buffer Overflow, Attacks on Wireless Networks, Identity theft, Foot printing and Social Engineering, Port Scanning, Email Investigation, E-Mail Tracking, IP Tracking, Email Recovery, Password Cracking.

Text Books:

1. Cryptography and Network Security, 3rd Edition, Behrouz A. Forouzan, Deb deep Mukhopadhyay, McGraw Hill, 2015.
2. SunitBelapure, Nina Godbole "Cyber Security: Understanding Cyber Crimes, Computer Forensics and Legal Perspectives,", WILEY, 2011.

Reference Books:

1. Network Security and Cryptography, First Edition, Bernard Menezes, Cengage Learning, 2018.
2. Cryptography and Network Security, William Stallings, Global Edition, 7e Pearson, 2017.

Web Resources:

1. https://archive.nptel.ac.in/courses/106/105/106105162/
2. https://ebooks.inflibnet.ac.in/csp11/chapter/introduction-to-network-security/
3. https://www.fortinet.com/resources/cyberglossary/what-is-cryptography
4. https://ischoolonline.berkeley.edu/cybersecurity/curriculum/cryptography/
5. https://www.mitel.com/articles/web-communication-cryptography-and-network-security
6. https://www.nist.gov/cybersecurity
7. https://www.codecademy.com/learn/introduction-to-cybersecurity

II Semester	L	T	P	C
	3	0	0	3
24MC2T03 :: OBJECT ORIENTED PROGRAMMING USING JAVA				

Course Objectives:

- To understand the basic concepts of object-oriented programming (OOP).
- To introduce the principles of inheritance and polymorphism, and demonstrate their relation to abstract classes.
- To understand the implementation of packages and interfaces.
- To introduce multithreading and exception handling concepts.
- To learn the design of Graphical User Interfaces (GUIs) using Swing controls.

Course Outcomes		Knowledge Level (K)#
CO1	Explain OOP concepts and basic Java programming skills, including data types and control statements.	**K2**
CO2	Apply inheritance, packages, and interfaces in Java to organize and modularize code.	**K3**
CO3	Analyze and implement exception handling and multithreading in Java for error management and concurrent tasks.	**K4**
CO4	Design event-driven programs and GUIs using AWT components and the delegation event model.	**K6**
CO5	Evaluate and use Swing for creating advanced Java GUIs with sophisticated components.	**K5**

UNIT I:

Basics of Object-Oriented Programming (OOP): Need for OO paradigm, viewing world-Agents, responsibility, messages, methods, classes, 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 lifetime of variables, arrays, operators, expressions, control statements, type conversion and costing, simple Java programs, classes and objects concepts of classes, objects, constructors methods, access control, this keyword, garbage collection, overloading method and constructor, parameter passing, recursion, string handling.

UNIT II:

Inheritance: Hierarchical abstractions, Base class, 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 interfaces, implementing interfaces, applying interface variables in interface and extending interfaces.

UNIT III:

Exception Handling and Multithreading: Concepts of exception handling, benefits of exception handling,

Termination or resumptive models, exception hierarchy, usage of try, catch, throws, finally, built-in exceptions, creating own exception subclasses. Differences between multithreading and multitasking, thread lifecycle, creating and synchronizing threads, daemon threads, thread groups.

UNIT IV:

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, buttons, canvas, scrollbars, text components, check boxes, checkbox groups, choices, list panes, scroll pane, dialogs, menu bar, graphics, layout managers ty[es-border, grid, flow, card, grid bag.

UNIT V:

Swing: Introduction, limitations of AWT, MVC architecture, components, containers, exploring Swing-JFrame and JComponent, icons and Labels, text fields, buttons-The JButton class, check boxes, Radio buttons, Combo boxes, Tabbed panes, scroll panes, Trees, Tables.

Text Books:

1. *Java: The Complete Reference* (7th Edition) by Herbert Schildt, TMH

Reference Books:

1. *Java: How to Program* (8th Edition) by Deitel&Deitel, PHI
2. *Introduction to Programming with Java* by S. Dean, TMH
3. *Introduction to Java Programming* (6th Edition) by Y. Daniel Liang, Pearson
4. *Core Java Vol. 1 & 2* (7th Edition) by Cay S. Horstmann& Gary Cornell, Pearson
5. *Big Java* (3rd Edition) by Cay S. Horstmann, Wiley
6. *Object-Oriented Programming through Java* by P. Radha Krishna, University Press

Webresources:

1. https://onlinecourses.nptel.ac.in/noc20_cs58/preview
2. https://www.iitk.ac.in/esc101/05Aug/tutorial/information/resources.html
3. https://docs.oracle.com/javase/tutorial/index.html
   https://www.javacodegeeks.com/best-java-programming-resources

II Semester	L	T	P	C
	3	0	0	3
24MC2T04 :: SOFTWARE ENGINEERING				

Course Objectives:

- To understand the nature of software development and software life cycle models.
- To learn methods for capturing, specifying, visualizing, and analyzing software requirements.
- To understand basics of testing, software quality assurance, and software configuration management.
- To provide correctness proofs for algorithms.

Course Outcomes		Knowledge Level (K)#
CO1	Describe the history and characteristics of software engineering and compare SDLC methodologies like Waterfall, Spiral, and Agile.	**K2**
CO2	Apply techniques to gather and analyze software requirements and create a Software Requirements Specification (SRS).	**K3**
CO3	Analyze design strategies and evaluate software design using metrics such as coupling and cohesion.	**K4**
CO4	Develop and execute test cases using various techniques to ensure software quality.	**K6**
CO5	Evaluate maintenance models and reengineering techniques to manage software evolution and upkeep.	**K5**

UNIT I:

Introduction: Software Engineering and its history, Software crisis, Evolving of a Programming System Product, characteristics of software, Brooks' No Silver Bullet, and Software Myths, **Software Development Life Cycles :** Software Development Process, code-and-fix, model Waterfall model, Evolutionary model, Incremental Implementation,, prototyping, Spiral Model, Software Reuse, Critical Comparisons of SDLC models. **An Introduction to Non- Traditional Software Development Process:** Rational Unified Process, Rapid Application Development, Agile Development Process Introduction, Agile-SCRUM Sprint, Review, Retrospective, Planning) , XP, KANBAN, SAFE Agile.

UNIT II:

Requirements: Importance of requirement Analysis, User needs, Software Features, and Software Requirements, **Classes of User Requirements: E**nduring and volatile, sub-phases of Requirement Analysis, Functional and Non-functional requirements, Barriers to Eliciting user requirements, The software requirements document SRS standards, Requirements Engineering, Case Study of SRS for a Real Time System.

Tools for Requirements Gathering: Document flow chart, Decision Table, Decision Tree. Introduction to non-traditional Requirements.

UNIT III:

Software Design: Goals of good software design, Design strategies and methodologies, Data-oriented software design. **structured design:** Structure chart, Coupling, Cohesion, Modular structure, Packaging, Object oriented design, Top-down and bottom-up approach, Design patterns, **Structured Analysis:** DFD, Data Dictionary,

software measurement and Metrics: Various size-oriented Measures: Halstead's software science, Function Point (FP) based measures, Cyclomatic Complexity Measures: Control flow graphs Development: Selecting a language, Coding guidelines,

UNIT IV:

Software Testing: Testing process, Design of test cases, Functional Testing: Boundary value analysis, Equivalence class testing, Decision table testing, Cause-effect graphing, Structural testing, path testing, Data flow and mutation testing, unit testing, integration and system testing, Debugging, Alpha & beta testing, testing tools & standards.

UNIT V:

Software Maintenance: Management of maintenance, Maintenance process, Maintenance models, Regression testing, Reverse engineering, Software reengineering, Configuration management, documentation.

Text Books:

1. *Software Engineering: A Practitioner's Approach* (9th Edition) by R. S. Pressman, McGraw Hill

Reference Books:

1. *Software Engineering* by K.K. Aggarwal & Yogesh Singh, New Age International Publishers (3rd Edition, 2007)
2. *Software Engineering* by Ian Sommerville, Addison Wesley (9th Edition, 2010)
3. *An Integrated Approach to Software Engineering* by PankajJalote, Narosa Publishing House (3rd Edition, 2007)

Web Resources:

1. https://onlinecourses.nptel.ac.in/noc23_cs122/preview
2. https://nptelvideos.com/course.php?id=444
3. https://softengbook.org/
4. https://www.coursera.org/learn/introduction-to-software engineering?msockid=39a584c9c8ac6773281697f5c91e6633

II Semester	L	T	P	C
	3	0	0	3
24MC2T05 :: ARTIFICIAL INTELLIGENCE				

Course Objectives:

1. To learn the basic state space representation and intelligent systems categorization.
2. To explore knowledge and reasoning mechanisms for building expert systems.
3. To become familiar with supervised and unsupervised learning models.
4. To design and develop AI and machine learning solutions using modern tools.

Course Outcomes		Knowledge Level (K)#
CO1	Describe the foundational concepts of AI, including its definition, problems, system components, applications, and intelligent agent types.	**K2**
CO2	Apply various search and optimization algorithms to solve complex AI problems.	**K3**
CO3	Analyze knowledge representation and reasoning techniques using logic and their application in knowledge-based agents.	**K4**
CO4	Compare agent architectures and evaluate multi-agent systems for different AI applications.	**K5**
CO5	Design and develop expert systems using their architecture, components, and techniques for knowledge acquisition and heuristics.	**K6**

UNIT I:

Introduction to AI: Definition, problems, system components of AI Program, Foundations of AI , Applications of AI, Current trends in AI, Intelligent Agents Anatomy, structure, Types.

UNIT II:

Problem Solving and Searching:

Solving problems by Searching: Problem Solving Agent, Formulating Problems. Uninformed Search Methods: Breadth First Search (BFS), Depth First Search (DFS), Depth First Iterative Deepening (DFID), Informed Search Methods-Greedy best first Search, A* Search, Memory bounded heuristic Search. Local Search Algorithms and Optimization Problems- Hill climbing search Simulated annealing and local beam search.

UNIT III:

Knowledge and Reasoning: Knowledge-based Agents, The Wumpus World, and propositional logic, **First-Order Logic**: Syntax and Semantics, Inference in FOL,, forward chaining, backward chaining, knowledge Engineering in First- Order Logic, Unification and Resolution.

UNIT IV:

Agents : Definitions of agents, Agent architectures (e.g., reactive, layered, cognitive), Multi- agent systems-Collaborating agents Competitive agents, Swarm systems and biologically inspired models. **Expert Systems:** Representing and Using Domain knowledge, Expert System Shells, Explanation, Knowledge Acquisition.

UNIT V:

Expert Systems: Architecture of expert systems, Roles of expert systems,, knowledge Acquisition, Meta knowledge, Heuristics. Expert systems- MYCIN, DART, XOON, Expert system shells.

Text Books:

1. *Artificial Intelligence* by Saroj Kaushik, Cengage Learning India, 2011
2. *Artificial Intelligence and Machine Learning* by Vinod Chandra S.S., Anand Hareendran S.
3. *Artificial Intelligence: A Modern Approach* (2nd Edition) by Stuart J. Russell and Peter Norvig, Pearson

Reference Books:

1. *PROLOG Programming for Artificial Intelligence* by Ivan Bratko, Pearson Education (3rd Edition)
2. *Artificial Intelligence* by Elaine Rich and Kevin Knight (3rd Edition)
3. *Data Mining Concepts and Techniques* by Han Kamber, Morgan Kaufmann Publishers
4. *Artificial Intelligence* by G. Luger and W. A. Stubblefield (3rd Edition), Addison-Wesley Longman, 1998

Web Resources:

1. https://onlinecourses.nptel.ac.in/noc22_cs56/preview
2. https://openlearning.mit.edu/news/explore-world-artificial-intelligence-online-courses-mit
3. https://cse.iitk.ac.in/users/cs365/2015/resources.html
4. https://microsoft.github.io/AI-For-Beginners/
5. https://artint.info/3e/resources/index.html
6. https://web.dev/explore/ai

II Semester (Program Elective-I)	L	T	P	C
	3	0	0	3
24MC2TE1 :: DESIGN AND ANALYSIS OF ALGORITHMS				

Course Objectives:

- To analyze the asymptotic performance of algorithms.
- To write rigorous correctness proofs for algorithms.
- To become familiar with major algorithms and data structures.
- To apply important algorithmic design paradigms and methods of analysis.
- To synthesize efficient algorithms for common engineering design situations.

Course Outcomes		Knowledge Level (K)#
CO1	Explain algorithm fundamentals and performance analysis, including space/time complexity and asymptotic notations.	**K2**
CO2	Apply divide and conquer and greedy methods to solve problems like sorting, matrix multiplication, and job sequencing.	**K3**
CO3	Analyze dynamic programming approaches to solve optimization problems such as matrix chain multiplication and the traveling salesperson problem.	**K4**
CO4	Develop solutions to combinatorial problems using backtracking techniques.	**K6**
CO5	Evaluate branch and bound strategies for complex problems and differentiate between NP-Hard and NP-Complete problems.	**K5**

UNIT I:

Introduction: Algorithm, Pseudo code for expressing algorithms, Performance Analysis Space complexity and Time complexity, Asymptotic Notation- Big Oh notation, Omega notation, Theta notation, Little oh notation, Probabilistic analysis Amortized analysis. Disjoint Sets- disjoint set operations, union and find algorithms, spanning trees, connected and bi-connected components.

UNIT II:

Divide and Conquer: General method, applications-Binary search, Quicksort, 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.

UNIT III:

Dynamic Programming: General method, applications-Matrix chain multiplication, Optimal binary search trees, 0/1 knapsack problem, All pairs shortest path problem, Traveling salesperson problem, Reliability design.

UNIT IV:

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

UNIT V:

Branch and Bound: General method, applications-Traveling salesperson 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.

Text Books:

1. *Fundamentals of Computer Algorithms* by Ellis Horowitz, SatrajSahni, and Rajasekaran, Universities Press
2. *The Algorithm Design Manual* (2nd Edition) by Steven S. Skiena, Springer
3. *Introduction to Algorithms* (2nd Edition) by T.H. Cormen, C.E. Leiserson, R.L. Rivest, and C.S. Stein, PHI Pvt. Ltd

Reference Books:

1. *Introduction to the Design and Analysis of Algorithms* by AnanyLevitin, PEA
2. *Design and Analysis of Algorithms* by ParagHimanshu Dave and HimansuBalachandra Dave, Pearson Education
3. *Introduction to Design and Analysis of Algorithms: A Strategic Approach* by R.C.T. Lee, S.S. Tseng, R.C. Chang, and T. Tsai, McGraw Hill
4. *Design and Analysis of Algorithms* by Aho, Ullman, and Hopcroft, Pearson Education

Web Resources:

1. https://nptel.ac.in/courses/106106131
2. http://openclassroom.stanford.edu/MainFolder/CoursePage.php?course=IntroToAlgorithms
3. https://ocw.mit.edu/courses/6-046j-design-and-analysis-of-algorithms-spring-2015/resources/ lecture-notes/
4. https://ocw.mit.edu/courses/6-046j-design-and-analysis-of-algorithms-spring-2015/resources/lecture-notes/
5. https://aofa.cs.princeton.edu/home/

II Semester (Program Elective-I)	L	T	P	C
	3	0	0	3
24MC2TE2 :: ADVANCED UNIX PROGRAMMING				

Course Objectives:

1. To understand the fundamental design of UNIX programming.
2. To become fluent with the system calls provided in UNIX.
3. To design and build applications/services over the UNIX operating system.

Course Outcomes		Knowledge Level (K)#
CO1	Describe UNIX utilities and develop basic shell scripts for file handling and process management.	**K2**
CO2	Handle UNIX files and directories using system calls and directory functions, and differentiate between system calls and library functions.	**K3**
CO3	Analyze UNIX process management and signal handling, including handling zombie and orphan processes.	**K4**
CO4	Implement inter-process communication techniques such as pipes and message queues for client-server programs.	**K6**
CO5	Evaluate the use of shared memory and sockets for client-server communication using TCP and UDP protocols.	**K5**

UNIT I:

Review of UNIX Utilities and Shell Programming: File handling utilities, security by file permissions, process utilities, disk utilities, networking commands, backup utilities. **Shell Programming:** Shell, Shell responsibilities, pipes and input redirections, output redirection, here documents, the shell as a programming language, shell metacharacters, shell variables, shell commands, the environment, control structures, shell script examples.

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, ioctl, umask, dup, dup2, Differences between system calls and library functions. **File and Directory Maintenance:** chmod, chown, unlink, link, symlink, mkdir, rmdir, chdir, getcwd. **Directory Handling System Calls:** opendir, readdir, closedir, rewinddir, seekdir, telldir.

UNIT III:

UNIX Process: Threads, and Signals: Process ,Process structure, starting new process, waiting for a process, zombie process, orphan process, process control, process identifiers, system calls for process management-fork, vfork, exit, wait, waitpid, exec, system. **Signals:** Signal functions, unreliable signals, interrupted system calls, kill, raise, functions, alarm, pause functions, abort, sleep functions **.**

UNIT IV:

Inter process Communication: Introduction to IPC, IPC between processes on a single computer system, IPC

between processes on different systems, pipes, FIFOs, message queues, semaphores and shared memory. Differences between pipes and FIFOs. Implementing a client server program using pipes and FIFOs. **Message Queue**s-:IPC permission issues, Access perm queues, client/server example **Semaphores**: Creating semaphore sets, Unix kernel support for semaphores, Unix APIs for semaphores, file locking using semaphores.

UNIT V:

Shared Memory and Sockets: Working with shared memory segments, UNIX kernel support shared memory, client/server example.

Sockets: Berkeley sockets, socket structure, socket system calls for connection oriented protocol and connectionless protocol, implementing client server programs using TCP and UDP sockets

Text Books:

1. *Advanced Programming in the UNIX Environment* (2nd Edition) by Richard Stevens, Pearson Education
2. *Unix Concepts and Applications* (3rd Edition) by Sumitabha Das, TMH

Reference Books:

1. *Unix and Shell Programming* by Sumitabha Das, TMH
2. *A Beginner's Guide to UNIX* by N.P. Gopalan& B. Siva Selva, PHI
3. *Unix Shell Programming* by Stephen G. Kochan& Patrick Wood
4. *Unix Shell Programming* (3rd Edition) by Lowell Jay Arthus& Ted Burns, Galgotia

Web Resources:

1. https://archive.nptel.ac.in/courses/117/106/117106113/
2. https://stevens.netmeister.org/631/
3. https://www.cs.fsu.edu/~asriniva/courses/aup02/lectures.html

II Semester (Program Elective-I)	L	T	P	C
	3	0	0	3
24MC2TE3 :: DATA WAREHOUSING AND DATA MINING				

Course Objectives:

- To be familiar with the mathematical foundations of data mining tools.
- To understand and implement classical models and algorithms in data warehousing and mining.
- To characterize patterns discovered through association rule mining, classification, and clustering.
- To develop skills in selecting appropriate data mining algorithms for practical problems.

Course Outcomes		Knowledge Level (K)#
CO1	Describe data mining concepts, including data types and quality, and use visualization techniques to explore datasets.	**K2**
CO2	Apply and evaluate classification techniques like decision trees and support vector machines to solve classification problems.	**K3**
CO3	Analyze and implement association rule mining to uncover patterns using algorithms like FP-Growth.	**K4**
CO4	Evaluate and compare clustering algorithms and use metrics to assess clustering effectiveness.	**K5**
CO5	Create and analyze web data mining processes and understand search engine functionality and ranking methods.	**K6**

UNIT I:

Introduction to Data Mining Types of Data, Data Quality, Data processing, Measures of Similarity and Dissimilarity, Exploring Data: Data Set, Summary Statistics, Visualization, Data Warehouse, OLAP, multidimensional data analysis.

UNIT II:

Classification: Basic concepts, Decision Trees, model evaluation, General approach for solving a classification problem, Decision Tree induction, Model overfitting: due to presence of noise, due to lack of representation samples, Evaluating the performance of classifier Nearest Neighbor classifier, Bayesian classifier, Support vector Machines: linear SVM, Separable and Non-Separable cases.

UNIT III:

Association Analysis: Problem definition, frequent item-set generation, rule generation, compact representation of frequent item sets, FP-Growth Algorithm, Handling Categorical Continuous attributes, Concept hierarchy, Sequential, Subgraph patterns.

UNIT IV:

Clustering: Overview, K-means, Agglomerative Hierarchical Clustering, DBSCAN, Cluster evaluation: overview, Unsupervised Cluster Evaluation using Cohesion, Separation, using Proximity Matrix, Scalable Clustering Algorithms.

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 Web Pages, Enterprise search.

Text Books:

1. *Introduction to Data Mining* by Tan, Steinbach, and Vipin Kumar, Pearson Education, 2016
2. *Data Mining: Concepts and Techniques* (2nd Edition) by Jiawei Han and Micheline Kamber, Elsevier
3. *Data Mining* by Vikram Pudi and P. Radha Krishna, Oxford University Press

Reference Books:

1. *Data Mining:* The Text book, Springer,May2015, CharuC. Aggarwal.

Web Resources:

1. https://nptel.ac.in/courses/106/105/106105174/
2. https://www.saedsayad.com/data_mining.htm
3. https://ocw.mit.edu/courses/15-062-data-mining-spring-2003/pages/lecture-notes/
4. https://www2.cs.uh.edu/~arjun/courses/dm/
5. https://www.rdatamining.com/resources/online-documents-books-and-tutorials
6. https://dataminingbook.info/book_html/

II Semester	L	T	P	C
	3	0	0	3
MOOCS-1(NPTEL/SWAYAM) (Recommended 12 week course with 3 credits)				

II Semester	L	T	P	C
	0	0	3	1.5
24MC2L01 :: OBJECT ORIENTED PROGRAMMING USING JAVA LAB				

Course Objectives:

1. To understand how to design, implement, test, debug, and document programs that use basic data types and computation, simple I/O, structures, string handling and functions.
2. To understand the importance of Classes & objects along with constructors, Arrays and Vectors.
3. Discuss the principles of inheritance, interface and packages and demonstrate through problem analysis assignments how they relate to the design of methods, abstract classes and interfaces and packages.
4. To understand the importance mechanisms.
5. To learn experience of designing, implementing, graphical user interfaces in Java using applet and AWT that respond to different user events.
6. To understand Java Swings for designing GUI applications based on MVC architecture.

List of Experiments:

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 that uses both recursive and non recursive functions to print the nth value of the Fibonacci sequence.
2. Write a Java Program that prompts the user for an integer and then prints out all the prime numbers up to that Integer.
3. Write a Java Program that checks whether a given string is a palindrome or not. Ex. MALAYALAM is a palindrome
4. Write a Java Program for sorting a given list of names in ascending order.
5. Write a Java Program that illustrates how runtime polymorphism is achieved.
6. Write a Java Program to create and demonstrate packages.
7. Write a Java Program, using StringTokenizer class, which reads a line of integers and then displays each integer and the sum of all integers.
8. Write a Java Program that reads a 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 contents using File Input Stream class Write a Java Program that displays the number of characters, lines and words in a text/text file.
9. Write a Java Program 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.
10. Write a Java Program 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. Write a Java Program for handling mouse events.
11. Write a Java Program demonstrating the life cycle of a thread.
12. Write a Java Program that lets users create Pie charts. Design your own user interface (with Swings & AWT).
13. Write a Java Program to implement a Queue, using user defined Exception Handling (also make use of throw, throws).

Web Resources:

1. https://www.iitk.ac.in/esc101/05Aug/tutorial/information/resources.html
2. https://labex.io/skilltrees/java
3. https://docs.oracle.com/javase/tutorial/index.html
4. https://introcs.cs.princeton.edu/java/home/

II Semester	L	T	P	C
	0	0	3	1.5
24MC2L02 :: NETWORKS AND SECURITY LAB				

Course Objectives:

- To learn basic understanding of cryptography, how it has evolved, and some key encryption techniques used today.
- To understand and implement encryption and decryption using Caesar Cipher, Substitution Cipher, Hill Cipher.

Course Outcomes		Knowledge Level (K)#
CO1	Implement data link layer framing methods like character and bit stuffing in C.	**K3**
CO2	Develop a C program to compute CRC checksums using CRC-16 and CRC-CCITT.	**K3**
CO3	Implement Dijkstra's algorithm in C or Java to find the shortest path in a graph.	**K3**
CO4	Calculate and present routing tables using the distance vector routing algorithm.	**K4**
CO5	Write Java programs for encryption and decryption using various algorithms and key exchange mechanisms.	**K6**

List of Experiments:

1. Implement the data link layer farming methods such as character stuffing and bit stuffing.
2. Implement a data set of characters the three CRC polynomial – CRC12, CRC 16 and CRC CCIP.
3. Implement Dijkstra ‘s algorithm to compute the Shortest path thru a graph.
4. Take an example subnet graph with weights indicating delay between nodes. Now obtain Routing table art each node using distance vector routing algorithm
5. Take an example subnet of hosts. Obtain a broadcast tree for it.
6. Write a C program that contains a string (char pointer) with a value ‘Hello World’. The program should XOR each character in this string with 0 and display the result.
7. Write a C program that contains a string (char pointer) with a value ‘Hello World’. The program should AND or and XOR each character in this string with 127 and display the result
8. Write a Java program to perform encryption and decryption using the following algorithms:
    a) Caesar Cipher
    b) Substitution Cipher
    c) Hill Cipher
9. Write a Java program to implement the DES algorithm logic
10. Write a C/JAVA program to implement the Blowfish algorithm logic.
11. Write a C/JAVA program to implement the Rijndael algorithm logic
12. Using Java Cryptography, encrypt the text “Hello world” using Blowfish.
13. Create your own key using Java key tool.
    a. Write a Java program to implement RSA Algorithm

b. Write a Java program to implement Public key Algorithm like El Gamal
c. Implement the Diffie-Hellman Key Exchange mechanism using HTML

Web Resources:

1. https://csrc.nist.rip/publications/nistpubs/800-12/800-12-html/chapter19.html
2. http://vlabs.iitkgp.ac.in/ant/
3. https://networklessons.com/labs/network-fundamentals-lab-1
4. https://elearn.daffodilvarsity.edu.bd/course/view.php?id=10230
5. https://www.cybrary.it/practice-lab/cryptography-basics
6. https://www.infosecinstitute.com/resources/cryptography/cryptographic-algorithms- lab/

II Semester	L	T	P	C
	0	0	3	1.5
24MC2L03 :: EMPLOYABILITY SKILLS-1 LAB				

Course Objectives:

- Improve analytical thinking, listening, and communication skills, including both verbal and non-verbal techniques.
- Develop self-management skills, etiquette, and verbal ability for effective personal and professional interactions.
- Equip students with job-oriented skills such as group discussions, resume preparation, and interview techniques for career success.

Course Outcomes		Knowledge Level (K)#
CO1	Demonstrate effective communication techniques and strategies for self-analysis and maintaining a positive attitude	
CO2	Apply self-management techniques and understand various etiquette practices for personal and professional contexts	
CO3	**Develop skills in note-making letter writing and verbal**	
CO4	Analyze and apply techniques for effective group discussions and evaluate personal performance through mock discussions	
CO5	Create a professional resume and demonstrate interview skills through mock interviews	

UNIT – I:

Analytical Thinking & Communication Skills: Analytical Thinking & Listening Skills: **Self-Introduction, Shaping Young Minds – A Talk by Azim Premji (Listening Activity), Self-Analysis, Developing Positive Attitude, Perception.** Communication Skills: Verbal Communication: **Techniques and practices .**Non-Verbal Communication: **Body Language**

Unit-II:

Self-Management & Etiquette: Self-Management Skills: **Anger Management, Stress Management, Time Management, Six Thinking Hats, Team Building, Leadership Qualities,** Etiquette: **Social Etiquette, Business Etiquette, Telephone Etiquette, Dining Etiquette**

Unit-III:

Standard Operation Methods & Verbal Ability: Standard Operation Methods: **Note Making, Note Taking, Minutes Preparation, Email & Letter Writing.** Verbal Ability: **Synonyms, Antonyms, One Word Substitutes, Correction of Sentences, Analogies, Spotting Errors, Sentence Completion, Course of Action, Sentence Assumptions, Sentence Arguments, Reading Comprehension, Practice Work**

UNIT – IV:

Job-Oriented Skills – I:Group Discussion: Techniques and strategies, Mock Group Discussions

UNIT – V:

Job-Oriented Skills – II: Resume Preparation: Key components and structure, Sample formats and best practices. **Interview Skills:** Interview Techniques, Mock Interviews

Text Books and Reference Books:

1. Barun K. Mitra, *Personality Development and Soft Skills*, Oxford University Press, 2011.
2. S.P. Dhanavel, *English and Soft Skills*, Orient Blackswan, 2010.
3. R.S. Aggarwal, *A Modern Approach to Verbal & Non-Verbal Reasoning*, S. Chand & Company Ltd., 2018.
4. Raman, Meenakshi& Sharma, Sangeeta, *Technical Communication Principles and Practice*, Oxford University Press, 2011.

Web References:

1. www.Indiabix.com
2. www.freshersworld.com