DEPARTMENT OF COMPUTER APPLICATIONS

MASTER OF COMPUTER APPLICATIONS

ACADEMIC REGULATIONS COURSE STRUCTURE AND DETAILED SYLLABUS

AR-26

ADITYA INSTITUTE OF TECHNOLOGY AND MANAGEMENT
(AN AUTONOMOUS INSTITUTION)
APPROVED BY AICTE,
RECOGNIZED UNDER 2(F) & 12(B) OF UGC
PERMANENTLY AFFILIATED TO JNTUGV, VIZIANAGARAM
K.KOTTURU, TEKKALI, SRIKAKULAM -532201, ANDHRA PRADESH.

Aditya Institute of Technology and Management

ACADEMIC REGULATIONS (AR26)

For

Master of Computer Applications (MCA) Programme

(Duration: Two Years)

(Applicable for the batches admitted from the A.Y. 2026-27)

ADITYA INSTITUTE OF TECHNOLOGY AND MANAGEMENT
(AUTONOMOUS)
Approved by AICTE, Accredited by NBA & NAAC,
Recognized under 2(f) and 12(b) of UGC
Permanently Affiliated to JNTUGV, Vizianagaram
K. Kotturu, Tekkali, Srikakulam-532201, Andhra Pradesh, India

ADITYA INSTITUTE OF TECHNOLOGY AND MANAGEMENT (AUTONOMOUS)
Approved by AICTE, Accredited by NBA & NAAC, Recognized under 2(f) and 12(b) of UGC and Permanently Affiliated to JNTUGV, Vizianagaram
K. Kotturu, Tekkali, Srikakulam-532201, Andhra Pradesh, India

---

ACADEMIC REGULATIONS

Applicable for the students of Master of Computer Applications (MCA) PG Programme admitted from the Academic Year 2026-27 onwards. The MCA Degree of Aditya Institute of Technology and Management shall be conferred on candidates who are admitted to the program and who fulfill 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:

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.

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

The minimum instruction days in each semester are 90.

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.

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
2 Hours Practical (Lab) per week	1 Credit

3. ATTENDANCE:

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

Condonation of shortage of attendance up to 10% (65% and above, and below 75%) maybe given by the College academic committee.

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

Shortage of attendance below 65% shall in NO case be condoned.

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

A stipulated fee shall be payable towards condonation of shortage of attendance.

4. EVALUATION:

The performance of the candidate in each semester shall be evaluated subject-wise with a maximum of 100 marks on the basis of continuous Internal Exams (40 marks) and End Semester Examination (60 marks).

A candidate shall be deemed to have secured the minimum academic requirement in a subject ifhe 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.

Theory Subject:

For theory subjects, during a semester, there shall be two midterm examinations. Each midterm examination shall be conducted for a duration of 120 minutes and contains five descriptive questions (without choice), each question for 10 marks, and it will be scaled down to 40 marks.

Weightage of Marks					
	No. of questions	**I MID**		**No. of questions**	**II MID**
I Unit	2	20 Marks	III Unit	1	10 Marks
II Unit	2	20 Marks	IV Unit	2	20 Marks
III Unit	1	10 Marks	V Unit	2	20 Marks

The first midterm examination will be conducted usually after completion of 50% syllabus (i.e. first 2.5 Units), and the second midterm examination will be conducted usually at the end of instructions (after completion of remaining 50% syllabus i.e. remaining 2.5 Units).

Internal marks can be calculated with 80% Weightage for better of the two midterm exams and 20% weightage for another midterm exam.

The end semester examinations (60 marks) shall be conducted by the examination for duration of 180 minutes and contains five descriptive questions, each question for 12 marks. 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.

Laboratory:

The internal marks for laboratory are 40 marks and the marks shall be awarded based on the day to day work - 10 marks, record - 10 marks and the remaining 20 marks to be awarded by conducting an internal laboratory test.

External marks for laboratory are 60 marks and 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 external examiner. Laboratory examination must be conducted with breakup of description (10 marks), programming code (20 marks), output (15 marks), and viva-voce (15 marks).

Internship/industry oriented mini project (minimum 4 weeks):

There shall be an internship/industry oriented mini project, one need to complete during summer vacations (either within or after II-semester).

Self study report for the Internship/report on mini project after II semester shall be submitted and evaluated during the III semester.

The Self study report for the Internship/report on mini project will be evaluated for a total of 100 marks consisting of 40 marks for internal assessment and 60 marks for semester-end examination.

Internal assessment shall be done by internal supervisor.

The semester-end examination (Viva-Voce) shall be conducted by the committee, consisting of an external examiner, head of the department and internal supervisor of the Internship/industry oriented mini project.

5. PROJECT/DISSERTATION WORK EVALUATION:

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

A PRC shall be constituted with Head of the Department and two other senior faculty

members (one will be the guide).

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

After registration candidate has to submit, in consultation with his project supervisor, the title, objective and plan of action of his project work to the PRC for approval. The student can initiate the Project work, only after obtaining the approval from PRC after the III semester end examinations. The duration of the project is for one semester.

If a candidate wishes to change his supervisor or topic of the project, he can do so with the approval of the PRC, however, the PRC shall examine whether or not the change of topic/supervisor leads to a major change of his initial plans of project proposal. If yes, his date of registration for the project work starts from the date of change of Supervisor or topic as the case may be.

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

A candidate is permitted to submit Project Thesis only after the approval of PRC. For the approval of PRC the candidate shall submit the draft copy of thesis to the Head of the Department and shall make an oral presentation before the PRC.

Three copies of the Project Thesis certified by the supervisor shall be submitted to the College along with plagiarism report (<50%).

The thesis shall be adjudicated by one examiner selected by the Principal. For this, the Head of the Department shall submit a panel of 3 examiners, eminent in that field, with the help of the guide concerned.

A board consisting of the Supervisor, Head of the Department and the examiner approved by the College who adjudicated the thesis shall conduct Viva-Voce examination. The Board shall jointly report the candidate's work for a maximum of 100 Marks and the candidate has to secure minimum of 50 marks for pass. The Head of the Department shall coordinate and make arrangements for the conduct of Viva-Voce examination.

6. MINIMUM ACADEMIC REQUIREMENTS:

Condition for pass and award of credits for a course:

A candidate shall be declared to have passed the individual course if he/she secures not less than **40%** of marks in the end examination (in fraction rounded to lower digit) and a minimum of **50%** of marks in the sum total of the mid semester and end examination marks taken together.

On passing a course of a program, the student shall earn assigned credits in that Course.

Method of Awarding Letter Grades and Grade Points for a Course:

A letter grade and grade points will be awarded to a student in all courses having credits, based on his/her performance as per the grading system given below.

% of marks	Level	Letter Grade	Grade Points
≥ 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	Absent	AB	0

Calculation of Semester Grade Points Average (SGPA):

The performance of each student at the end of each semester is indicated in terms of SGPA.

The SGPA is calculated as below:

$$\text{SGPA} = \frac{\Sigma(\text{CR} \times \text{GP})}{\Sigma\text{CR}}$$ (for all courses passed in a semester)

Where CR = Credits of a Course
GP = Grade points awarded for a course

Calculation of Cumulative Grade Points Average (CGPA):

The CGPA is calculated as below:

$$\text{CGPA} = \frac{\Sigma(\text{CR} \times \text{GP})}{\Sigma\text{CR}}$$ (for the entire program)

Where CR = Credits of a course
GP = Grade points awarded for a course

Both SGPA and CGPA shall be rounded off to 2 decimal points and reported in the transcripts.

Award of Class:

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

CGPA to Percentage Conversion Formula: (CGPA – 0.5) x 10

6. SUPPLEMENTARY EXAMINATIONS:

Supplementary examinations will be conducted along with the subsequent batches regular examinations.

7. WITHHOLDING OF RESULTS:

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

8. TRANSITORY REGULATIONS:

Discontinued or detained candidates are eligible for readmission (within the duration as mentioned in item 2) and when next offered.

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

9. GENERAL:

Wherever the words he/him/his occurs in the regulations, they include she/her/hers.

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

In the case of any doubt or ambiguity in the interpretation of the above rules/regulations, the decision of the Principal is final.

The College 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 College.

****

DISCIPLINARY ACTION FOR MALPRACTICES/IMPROPER CONDUCT IN EXAMINATIONS

	Nature of Malpractices/Improper conduct	Punishment
1	If the student 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 student 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.
	If the student gives assistance or guidance or receives it from any other student orally or by any other body language methods or communicates through cell phone with any student or students in or outside the exam hall with respect to any matter	Expulsion from the examination hall and cancellation of the performance in that subject only. In case of an outsider, he will be handed over to the police and a case is registered against him.
2	If the student 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 student is appearing	Expulsion from the examination hall and cancellation of the performance in that subject and all other subjects the student 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	If the student impersonates any other student in connection with the examination	The student who has impersonated shall be expelled from examination hall. The student is also debarred and forfeits the seat. The performance of the original student, who has been impersonated, shall be cancelled in all the subjects of the examination (including practical's and project work) already appeared and shall not be allowed to appear for examinations of the remaining subjects of that semester/year. The student is also debarred for two consecutive semesters from class work and all semester-end examinations. The continuation of the course by the student is subject to the academic regulations in connection with forfeiture of the seat. If the imposter is an outsider, he will be handed over to the police and a case is registered against him.
4	If the student smuggles the answer book or additional sheet or takes out or arranges to send out the question paper 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 student 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 student is also debarred for two consecutive semesters from class work and all semester-end examinations. The continuation of the course by the student is subject to the academic regulations in connection with forfeiture of the seat.
5	If the student uses objectionable, abusive or offensive language in the answer script or in letters to the examiners or writes to the examiner requesting him to award pass marks	Cancellation of the performance in that subject.
6	If the student 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	In case of students of the college, they shall be expelled from examination hall and cancellation of their performance in that subject and all other subjects the candidate(s) has (have) already

	walkout or instigates others to walk out or threatens the officer-in charge or any person on duty in or outside the examination hall or causes any injury to any of his relatives either by words 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 relatives, or indulges in any other act of misconduct or mischief which results in damage 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	appeared and shall not be permitted to appear for the remaining examinations of the subjects of that semester/year. The students also are debarred and forfeit their seats. In case of outsiders, they will be handed over to the police and a police case is registered against them.
7	If the student leaves the exam hall taking away answer script or intentionally tears off 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 student 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 student is also debarred for two consecutive semesters from class work and all university examinations. The continuation of the course by the candidate is subject to the academic regulations in connection with forfeiture of the seat.
8	If the student possesses any lethal weapon or firearm in the examination hall	Expulsion from the examination hall and cancellation of the performance in that subject and all other subjects the student 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 student 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 clauses 6, 7, 8	In case of student of the college, expulsion from the examination hall and cancellation of the performance in that subject and all other subjects the student has already appeared including practical examinations and project work and shall not be permitted for the remaining examinations of the subjects of that semester/year. The candidate is also debarred and forfeits the seat. Person(s) who do not belong to the college will be handed over to police and. a police case will be registered against them.
10	If the student 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 student 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 student has already appeared including practical examinations and project work of that semester/year examinations.

Ragging

Ragging is prohibited under the follwing Acts

Supreme Court in SLP No. 24295 of 2006 dt. 16-05-2007
High Court Judgement dt. 11-09-1997 in W.P.No. 26132/96
Act 26 of A.P. Legislative Assembly dt. 19-08-1997
Indian Parliament Notice dt. 26-07-2008
as per AICTE Notice dt. 01-07-2009
as per UGC Notice dt. 22-10-2009

Ragging within or outside any educational institution is Prohibited.

Ragging means doing an act which causes or is likely to cause Insult or Annoyance or Fear or Apprehension or Threat or Intimidation or outrage of modesty or Injury to a student.

	Imprisonment upto		Fine upto
Teasing Embarrassing and Humiliation	6 Months	+	Rs.1000/-
Assaulting or using criminal force or criminal intimidation	1 Year	+	Rs.2000/-
Wrongfully restraining or confining or causing hurt	2 Years	+	Rs.5000/-
Causing grievous hurt, kidnaping or rape or committing unnatural offence	5 Years	+	Rs.10,000/-
Causing death or abetting suicide	10 Years	+	Rs.50,000/-

Master of Computer Applications

Course Structure &Syllabus

MCA I Semester

S.No	CourseCode	Course Name	L	T	P	Credits
1	26MCA1001	Statistical Methods	3	0	0	3
2	26MCA1002	Computer Organization	3	0	0	3
3	26MCA1003	Operating Systems	3	0	0	3
4	26MCA1004	C & Data Structures	3	0	0	3
5	26MCA1005	Software Engineering	3	0	0	3
6	26MCA1101	Business Communication Lab	0	1	2	2
7	26MCA1102	Data Structures Using C Lab	0	0	3	1.5
8	26MCA1103	Statistics with R programming Lab	0	0	3	1.5
Total			15	0	6	20

MCA II Semester

S.No	CourseCode	Course Name	L	T	P	Credits
1	26MCA1006	Database Management Systems	3	0	0	3
2	26MCA1007	Computer Networks	3	0	0	3
3	26MCA1008	Object Oriented Programming with Java	3	0	0	3
4	26MCA1009	Design and Analysis of Algorithms	3	0	0	3
5	26MCA1010	Python Programming	2	0	0	2
6		**Elective-I**	3	0	0	3
	26MCA1E11	No SQL Databases				
	26MCA1E12	Mobile Application Development				
	26MCA1E13	Artificial Intelligence				
	26MCA1E14	Computer Vision and Image Processing				
	26MCA1E15	MOOC COURSE				
7	26MCA1104	Database Management Systems Lab	0	0	3	1.5
8	26MCA1105	Computer Networks Lab	0	0	3	1.5
9	26MCA1106	Java Programming Lab	0	0	3	1.5
10	26MCA1107	Python Programming Lab	0	0	3	1.5
Total			17	0	12	23
Internship/industry oriented mini project(4Weeks) (Mandatory) is to be carried out during summer vacation .It is evaluated at the end of III Semester						

AR26-MCA **I Year I Semester**

STATISTICAL METHODS

L	T	P	C
3	0	0	3

Code: 26MCA1001

Course Objectives:

- Introduce and describe Binomial, Poison, Normal distribution.
- Perform sampling theory and estimation.
- Perform large samples tests–test of means and proportion.
- Perform the small sample tests-t-test, F-test and Chi-square test and ANOVA for the given data.
- Determine correlation and regression coefficients for given data**.**

Course Outcomes:

After successful completion of the course, the student will be able to:

1. Apply Binomial, Poisson, and Normal distributions to calculate probabilities of random variables.
2. Analyze and apply sampling distributions and the Central Limit Theorem to estimate parameters using point and interval estimation.
3. Apply and analyze large sample hypothesis tests for population means and proportions.
4. Apply and analyze small sample tests such as *t-test, F-test, Chi-square test,* and ANOVA for given data.
5. Apply and analyze correlation and regression techniques to study relationships between variables.

UNIT–I: Probability Distributions: Introduction to Random Variables-Discrete distributions: Binomial, Poisson distributions. Continuous distribution: Normal distribution.

UNIT–II: Sampling distributions and Estimation: Introduction-Population and sample-Sampling distribution of means(σknown)-Central limit theorem-Point estimation-Maximum error of estimate-Interval estimation.

UNIT– III: Tests of Hypothesis (Large samples): Introduction –Hypothesis-Null and Alternative Hypothesis- Type I and Type II errors –Level of significance, Power of the test-One tail and two-tail tests-Tests concerning single mean and single proportion-Tests concerning difference of means and difference of proportions.

UNIT- IV: Test of Hypothesis (Small Samples): Introduction– student's-t–test, F-test, Chi-square Test-ANOVA for one-way and two-way classified data.

UNIT- V: Correlation and Regression: Introduction-Concept of correlation–types of correlation-Karl-Pearson correlation coefficient method and its properties-Rank Correlation Coefficient-Regression-Linear regression and its properties.

Text Books:

1. Richards AJohnson,IrvinMillerandJohnsonEFreund.“ProbabilityandStatisticsfor Engineering”,9th Edition,PHI.
2. G.JayKerns,“IntroductiontoProbabilityandStatisticsUsingR”, 3rd Edition, CRC Press, 2018.
3. T.K.V.Iyengar, Dr. B.Krishna Gandhi, S,Ranganathamand Dr.M.V.S.S.N. Prasad, “ProbabilityandStatistics”, SChand Publications.

ReferenceBooks:

1. SharonL.Myers,KeyingYe,RonaldEWalpole,“ProbabilityandStatisticsEngineersandthe Scientists”, 9th Edition,Pearson ,2024.
2. WilliamMendenHall,RobertJ.BeverandBarbaraBever,“Introductiontoprobabilityand statistics”,Cengagelearning, 2009.
3. JohannesLedolterandRobertV.Hogg,“AppliedstatisticsforEngineersandPhysical Scientists”,3rd Edition,Pearson, 2010.
4. ErwinKreyszig,“AdvancedEngineering Mathematics”,10thEdition,Wiley-India.
5. Sheldon,M.Rosss,“IntroductiontoprobabilityandstatisticsEngineersandtheScientists”, 4th edition,AcademicFoundation, 2011.

AR26-MCA **I Year I Semester**

Computer Organization

Subject Code: 26MCA1002

L	T	P	C
3	0	0	3

Course Objectives:

- To understand Principles and the Implementation of Computer Arithmetic.
- To learn Operations of CPU including RTL, ALU, Instruction Cycle and Busses.
- To understand Fundamentals of different Instruction Set Architectures and their relationship to the CPU Design.
- To understand Memory System and I/O Organization.
- To learn Principles of Multiprocessor Systems.

Course Outcomes:

The student will be able to:

1. Understand the architecture of modern computer, and also understanding of how the computer performs arithmetic operations on positive and negative numbers.
2. Understand of different register transfers and instruction types.
3. Develop a detailed understanding of architecture and functionality of central processing unit.
4. Exemplify in a better way the memory organization is communicating with processing unit.
5. Understand of I/O devices communicating with Processing Unit and also knowing thecharacteristics of multi processors

UNIT -I:

Basic Structure of Computers: Functional unit, Basic Operational concepts, Busstructures, System Software, Performance, The history of computer development.

UNIT -II:

Machine Instruction and Programs: Instruction and Instruction Sequencing: RegisterTransfer Notation, Assembly Language Notation, Basic Instruction Types, BasicInput/output Operations, The role of Stacks and Queues in assembly language programming. Component of Instructions: Logic Instructions, shift and Rotate Instructions.

UNIT -III:

Type of Instructions: Arithmetic and Logic Instructions: Increment, decrement, add, subtract, multiply, divide, addwith carry, subtract with borrow, AND, OR, XOR, Branch Instructions Addressing Modes: Implied, Immediate, Register, Register Indirect, Auto increment and Auto decrement, Direct address, Indirect Address, Relative Address, Indexed, Base Register addressing modes.

UNIT -IV:

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

UNIT -V:

The MEMORY SYSTEMS: Basic memory circuits, Memory System Consideration, Read-Only Memory: ROM, PROM, EPROM, EEPROM, Flash Memory, Cache Memories: Mapping Functions, Processing Unit: Fundamental Concepts: Register Transfers, performing an Arithmetic or Logic Operation, Micro programmed Control: Microinstructions, Micro program Sequencing, Wide Branch Addressing Microinstructions with next –Address Field.

TEXT BOOKS:

1. Computer Organization, Carl Hamacher, ZvonksVranesic, SafeaZaky, 5th Edition, McGraw Hill.

2. Computer Architecture and Organization, John P. Hayes ,3rd Edition, McGraw Hill.

3. Computer System Architecture, M. Morris Mano, Rajib Mall,3/e

REFERENCE BOOKS:

1. Computer Organization and Architecture – William Stallings SixthEdition, Pearson/PHI

2. Structured Computer Organization – Andrew S. Tanenbaum, 4th Edition PHI/Pearson

3. Fundamentals or Computer Organization and Design, -Sivaraama Dandamudi Springer Int. Edition.

WEB LINKS:

https://www.geeksforgeeks.org/computer-organization-architecture/computer-organization-and-architecture-tutorials/

https://www.udemy.com/topic/computer-organization/?srsltid=AfmBOorBLZBHWyXWQAbO1S-thRv95XOjNWI_zF77lYEDIRiVXPPNNGvL

AR26-MCA **I Year I Semester**

OPERATING SYSTEMS

Subject Code: 26MCA1003

L	T	P	C
3	0	0	3

Course Objectives:

- To understand how modern operating systems, work and how their different parts work together.
- To learn concepts like processes and threads, CPU scheduling, deadlocks, memory management, and file systems.
- To understand how memory is managed, including memory allocation and virtual memory.
- To learn how the operating system controls system resources such as processes, memory, files, and input/output devices.
- To introduce security and protection methods used to keep the operating system safe.

Course Outcomes:

By the end of this course the student will be able to

1. Identify the generations and core functionalities of operating systems.
2. Explain processes and threads, CPU scheduling, IPC, and synchronization techniques.
3. Analyze different memory management techniques.
4. Discuss file system access methods and file allocation methods.
5. Evaluate deadlock handling and disk scheduling algorithms

UNIT - 1:

Operating System structures: Overview of Operating System, Operating system functions, types of operating systems, system calls, types of system calls.

Process Management: Process States, Process Control Block (PCB), operations on processes, Process Scheduling Criteria, CPU Scheduling Algorithms (FCFS, SJF, Priority, Round Robin).

UNIT - 2:

Threads: Overview, Multithreading Models, Thread Libraries, Java Threads, Threading Issues, OS Examples.

Process Synchronization: The Critical- section problem, Petersons solution, Synchronization Hardware, Semaphores, Classic problems of Synchronization, Monitors, Atomic Transactions.

UNIT - 3:

Memory management: Main memory: Swapping, Contiguous memory Allocation, Paging, Structure of the Page table, Segmentation

Virtual memory: Virtual memory, Demand paging, Page Replacement, Allocation of frames

Page replacement algorithms:

FIFO, LRU and Optimal page replacement, Thrashing.

UNIT - 4:

File-system Interface: Concept, Access Methods, Directory structure, File system Mounting, Filesharing, Protection

File-system Implementation: File system structure, file system implementation, directory implementation, allocation methods: contiguous allocation, linked allocation and indexed allocation, free-space management.

UNIT - 5:

Deadlocks: System model, deadlock characterization, deadlock prevention, deadlock detection, deadlock avoidance, recovery form deadlock

Mass-storage structure: Overview of Mass-storage structure, Disk structure, disk attachment, disk scheduling.

Text Books:

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

2. Operating systems, 6/E, William Stallings, PHI/Pearson.

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

Reference Books:

1. Operating systems 3/e, Dietal, Dietal, Pearson.

2. An introduction to Operating systems, Concepts and practice, Pramod Chandra P. Bhat, PHI,5/e

Reference Link:

https://nptel.ac.in/courses/106/105/106105214/

AR26-MCA **I Year I Semester**

C & Data Structures

Subject Code: 26MCA1004

L	T	P	C
3	0	0	3

Course Objectives:

- To introduce the fundamentals of the C programming language and develop basic program design and implementation.
- To understand the fundamentals of data structures and implement linked list.
- To explore, understand and implement stack and queue data.
- To understand and learn different sorting techniques.
- To get the knowledge and learn the concepts and operations of tree data structure.

Course Outcomes:

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

1. Implement basic programs by using C concepts.

2. Understand Functions, Structures, Unions, Pointers, and File Handling in C for efficient memory management and data manipulation.

3. Select the data structures that efficiently model the information in a problem and Demonstrate the advantages of dynamic memory allocation via linked lists

4. Illustrate the applications of Stacks and Queues.

5. Implement and know the application of algorithms for sorting and nonlinear data structures

Unit-I

Introduction to C: Variables and Data Types, Operators and Expressions, Managing Input and Output operators, Decision making-branching and looping, Arrays.

Unit-II

Data structure: Definition, types of data structures Recursion Definition, Linear and binary recursion. Preliminaries of algorithms, analysis and complexity.

Linked list: singly linked list, Double linked list and circular linked list - implementation, insertion, deletion and searching operations on linear list.

Unit-III

Stacks-Operations, array and linked representations of stacks, stack applications.

Queues- operations using arrays and operations using linked list representations.

Unit-IV

Sorting Techniques: Insertion sort, selection sort, exchange-bubble sort, quick sort and merge sort Algorithms.

Unit-V

Trees: Binary Trees, terminology, representation and traversals- pre, post & in order traversals. Search Trees: Binary Search Trees, DFS and BFS, Shortest-Path Algorithm: Dijkstra's Algorithm.

Text Books:

1. Data Structures Using C. 2nd Edition, ReemaThareja, Oxford
2. Data Structures and Algorithm Analysis in C, 2ndedition, Mark Allen Weiss

Reference Books:

1. Data Structures: A Pseudo code Approach with C, 2nd Edition, R. F. Gilbergand B.A. Forouzan, Cengage Learning.
2. Classic data structures,2nd edition, Debasis Samantha, PHI.
3. Data Structures (Using C) 1st Edition, 2024 by Dr. Rajiv Chopra.

Reference Link:

https://nptel.ac.in/courses/106103069

AR26-MCA **I Year I Semester**

SOFTWARE ENGINEERING

Subject Code: 26MCA1005

L	T	P	C
3	0	0	3

Course Objectives:

- To understand Software Requirements and prepare Requirements Document.
- To understand the software development process models
- To learn Software design principles
- To learn about testing strategies
- To understand Software metrics

Course Outcomes:

1. Understand software requirements that form the background to develop complex software systems.

2. Apply an effective software engineering process, based on knowledge of widely used development models.

3. Translate requirements specification into an implement able design.

4. Identify a testing strategy for a software system, employing techniques such as black box and white box testing strategies.

5. Evaluate the quality of the requirements, analysis and design work during the module.

Unit – I

Introduction to Software Engineering: The evolving role of software, Changing Nature of Software, software myths. Software Requirements: Functional and non-functional requirements, User requirements, System requirements, Interface specification, the software requirements document. Requirements engineering process: Feasibility study, Requirement Elicitation and analysis, Requirements validation and Requirements Management

Unit – II

Process models: The waterfall model, Incremental process models: The Incremental model, The RAD model, Evolutionary process models: Prototyping model, The Spiral model, The concurrent development model, The Unified process model, Agile Process Model

Unit – III

Design Engineering: Design process and Design quality, Design concepts, the design model. Creating an architectural design: Software architecture, Data design, Architectural styles and patterns, Architectural Design. Performing User interface design: Golden rules, User interface analysis and design, interface analysis, Interface design steps.

Unit – IV

Testing Strategies: Verification and Validation, Unit Testing, Integration Testing, Validation testing, System testing, the art of Debugging, White-Box Testing: Basis Path Testing, Control

Structure Testing, Black-Box Testing: Equivalence Class Partitioning, Boundary Value Analysis, Orthogonal Testing, Combinatorial Testing, Pair-Wise Testing.

Unit – V

Productmetrics:SoftwareQuality,MetricsforAnalysisModel,ArchitecturalDesignMetrics Metrics for source code Empirical Estimation: COCOMO II Model QualityManagement:Qualityconcepts,SoftwareQualityAssurance(SQA),SoftwareReviews, Formal Technical Reviews(FTR), Statistical Software Quality Assurance(SSQA) ,Software Reliability,Software Security and Privacy

Text Books:

1. Roger S Pressman-Software Engineering,8th Edition, Tata McGraw Hill Education 2017
2. Sommerville, Software Engineering,7thEdition, PearsonEducation,2005

Reference Books:

1. Software Engineering- K.K. Agarwal &Yogesh Singh, NewAge International Publishers
2. Software Engineering: Chandramouli Subramanian, SaikatDutt, ChandramouliSeetharaman, B G Geetha- Pearson Education

Reference Link:

http://nptel.ac.in/courses/106101061/

AR26-MCA **I Year I Semester**

BUSINESS COMMUNICATION LAB

L	T	P	C
0	1	2	2

Subject Code: 26MCA1101

Course Objectives:

- To develop effective verbal and non-verbal communication skills required in professional and corporate environments.
- To improve the students' listening, speaking, reading, and writing (LSRW) skillsfor clear and effective communication.
- To enhance confidence in public speaking, including presentations, group discussions, and interviews.
- To develop professional writing skills such as email writing, business letters, and reports.
- To prepare students for workplace communication and career opportunities through resume writing, mock interviews, and corporate etiquette training.

Course Outcomes:

- Students will be able to communicate effectively in professional and workplace situations.
- Students will demonstrate improved listening, speaking, reading, and writing (LSRW) skills in English.
- Students will be able to deliver presentations and participate confidently in group discussions and interviews**.**
- Students will be able to prepare professional documents such as emails, business letters, and reports.
- Students will exhibit appropriate workplace communication and professional etiquette required in the corporate and IT industry.

Course Syllabus:

Unit – I: Fundamentals of Communication

Introduction to Communication, Process and Types of Communication (Verbal, Non-verbal, Written), Barriers to Communication, Listening Skills and Active Listening Techniques, Pronunciation practice and phonetic awareness, Basic speaking practice: self-introduction, introducing others

Lab Activities

- Self-introduction practice
- Listening exercises
- Pronunciation drills
- Group discussions on simple topics

Unit – II: Professional Speaking Skills

Formal and Informal Communication, Public Speaking Techniques, Presentation Skills, Group Discussion Skills, Interview Skills and Confidence Building

Lab Activities

- Mock presentations using PPT
- Group discussions on current topics
- Extempore speaking
- Mock interviews

Unit – III: Business Writing Skills

Principles of Effective Business Writing, Email Writing, Business Letters (Enquiry, Complaint, Order, Reply),Memo and Notice Writing, Report Writing Basics

Lab Activities

- Drafting professional emails
- Writing business letters
- Preparing short reports
- Editing and proofreading exercises

Unit – IV: Workplace Communication

Telephone and Virtual Communication,Meeting Etiquette and Participation,Corporate Etiquette and Professional Behavior,Interpersonal Communication in Teams,Conflict Resolution Communication

Lab Activities

- Role plays (office situations)
- Telephone conversation practice
- Conducting mock meetings
- Team communication exercises

Unit – V: Career Communication Skills

Resume / CV Writing,Cover Letter Writing,LinkedIn and Professional Profile Building,Technical Presentation Skills,Placement and HR Interview Preparation

Lab Activities

- Resume preparation workshop
- Mock HR and Technical Interviews
- Technical seminar presentations
- Peer feedback sessions

References:

1. Business Communication Today – Courtland L. Bovee and John V. Thill, Pearson Education.
2. Business Communication: Building Critical Skills – Kitty O. Locker and Donna S. Kienzler, McGraw-Hill Education.
3. Effective Technical Communication – M. Ashraf Rizvi, McGraw-Hill Education.
4. Communication Skills – Sanjay Kumar and PushpLata, Oxford University Press.
5. Basic Communication Skills for Technology – Andrea J. Rutherfoord, Pearson Education.

AR26 – MCA **I Year I Semester**

Data Structures using C Lab

Subject Code: 26MCA1102

L	T	P	C
0	0	3	1.5

Course Objectives

- To Design and implement various data structures.
- Implement operations like searching, insertion, and deletion, traversing mechanism.
- Apply suitable data structure for real world problems.
- Develop program for implementing various sorting techniques.

Course Outcomes

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

1. Implement various basic data structures and its operations.
2. Apply sorting and searching algorithms to given numbers
3. Implement various tree operations.
4. Implement various graphs algorithms.
5. Develop applications using various data structures.

List of Experiments

Experiment 1:

a. Writea C program to implement stack operations using arrays.
b. Write a C program for Evaluation of Postfix Expression Using Stack.
c. Write a C program for Conversion of Infix to Postfix Expression Using Stack.

Experiment 2:

Write C program to implement stack operations using Linked list.

Experiment 3:

Write C program to implement Queue operations using arrays.

Experiment 4:

Write C program to implement Queue operations using linked lists.

Experiment 5:

Write a C program to create a singly linked list and perform various operations on it.

Experiment 6:

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

Experiment 7:

Write a program to implement of Binary Search trees- Insertion and deletion.

Experiment 8:

Write a C program to implement Bubble sort.

Experiment 9:

Write a C program to implement Quick sort, to sort a given list of integers.

Experiment 10:

Write a C program to implement insertion sort.

Experiment 11:

Write a C program to implement merge sort.

Experiment 12:

Write a C program to implement BFS.

Experiment 13:

Write a C program to implement DFS.

Reference Books:

1. Data Structures and Algorithm Analysis in C, 2nded, Mark Allen Weiss.

AR26-MCA **I Year I Semester**

Statistics with R Programming Lab

Code: 26MCA1103

L	T	P	C
0	0	3	1.5

CourseObjectives:

- IntroducebasicRoperationsanddeterminetheprobabilitiesofarandomvariable using Binomial, Poisson's, Normal distribution.
- Determinetheprobabilitiesofsamplemeanusingcentrallimittheoremandto estimate confidenceintervalusingR programming.
- Perform z-test forsamplingdistribution usingR-programming.
- Analyze t-test and F-test forsamplingdistribution usingR-programming.
- Perform Chi-square test and Analysis of variance (ANOVA)usingR- programming.
- Calculatecorrelation and regression forgiven data usingR programming.

Course Outcomes:

On completion of this course, students will be able to:

1. ObtainprobabilitiesofrandomvariableofaBinomial,Poisson's,Normaldistribution

2. Calculate probabilities of sample mean and confidence interval

3. Perform Z– test,T– test and F-test forsamplingdistributions .

4. Analyze tests ofsignificanceusing Chi-squaretestand ANOVA.

5Determinethecoefficientofcorrelationandregressionequationsforgiven

1. Descriptive Statistics

Experiment 0: Introduction to R-programming

Experiment 1: WritethecommandsonRconsoletocalculatetheprobabilityof random variable forBinomial distribution functions.

Experiment 2: WritethecommandsonRconsoletocalculatetheprobabilityof random variable forPoison distribution functions.

Experiment 3: WritethecommandsonRconsoletocalculatetheprobabilityof random variable forNormal distribution functions.

Experiment 4: Write thecommandson R console toperform Correlation forgiven data.

Experiment 5: Writethecommandson Rconsoletocalculateprobabilityofmeanof random variable using central limittheorem.

2. Inferential Statistics

Experiment 6: Write the commandsofRconsoletocalculateconfidenceintervals forproportions and means

Experiment 7: WritethecommandsonRconsoletoperformz-testfortestingsingle proportion and differenceof proportions at αlevel of significance.

Experiment 8: WritethecommandsonRconsoletoperformz-testfortestingsingle mean and differenceofmeans at αlevel of significance.

Experiment 9: WritethecommandsonRconsoletoperformt-testfortestingthe single meanand differenceof means at αlevel ofsignificance.

Experiment 10: WritethecommandsonRconsoletoperformF-testfortestingthe equalityof populationvariancesat α level of significance.

Experiment 11: WritethecommandsonRconsoletoperform Chi-square testfor testing the goodness of fit and independence of attributes.

3. Prescriptive Statistics

Experiment 12: WritethecommandsonRconsoletoperformANOVAofoneway and two way classificationstoteston the basisof sample observations whether themeans of 3 or morepopulations areequal or not.

4. Predictive Statistics

Experiment 13: Write thecommandsonRconsoletoperformregression forgiven data.

Text Books:

1. DVLN Jogiraju, Agatamudi Lakshmana Rao, R. Visweswararao, Nagendra Kumar Bhupathi, "Computational Statistics and R Programming", Kalyani Publishers, 1st Edition, New delhi, India.

2. Thomas Mailund, Beginning Data Science in R: "Data Analysis, Visualization, and Modelling for the Data Scientist", Apress, Aarhus, Denmark.

Reference Books:

1. Dr. Agatamudi Lakshmana Ra, Dr. Ch. GanapathiSwamy, Sri N. Lakshmana Rao, Dr. RajamahanthiSanthikumar, "Probability & Statistics with R- Programming", Infinite Research, Registered Under MSME Government of India, UDYAM-AP-02-0028189, India.

2. Emmanuel Paradis, "R for Beginners", Institut Des Sciences De L'evalutionUniversite Montpellier II, F-34095 Montpellier cdex 05 France.

AR26-MCA **I Year II Semester**

Database Management Systems

Subject Code:26MCA1006

L	T	P	C
3	0	0	3

Course Objectives

- To introduce basic RDBMS concepts, SQL,
- To understand and apply database design principles using Entity–Relationship (ER) diagrams for modeling and developing efficient relational databases.
- To develop the ability to write and execute SQL queries for creating, retrieving, updating, and managing data in relational databases.
- To introduce transaction processing, issues and techniques relating to concurrency and recovery in multi-user database environments,
- To understand various Data structures for External Data storage and efficient retrieval

Course Outcomes

Students will be able to:

1. Differentiate Database Systems from File Systems and Define the Terminology, Features, Classifications, Characteristics embodied in Database Systems.
2. Interpret, Design and Implement an E-R Model.
3. Create /Modify the Structure and write optimized SQLQueries to extract and modify Information from Tables or Views.
4. Apply proper Techniques such as Normalization and analyze the applicability of a Specific Normal form in designing a Database.
5. Explain broad range of Database Management issues including Data integrity, Concurrency and Recovery and Compare Various Indexing, Hashing and File Organization Techniques.

Unit – I

Database System Applications; Database Systems versus file Systems; View of Data : Data Abstraction, Instances and Schemas ; Data Models :The ER Model ,Relational Model, Other Data Models ;Database Languages: DDL , DML ,Database Access from Application Programs; Data base Users and DBA ;Database System Structure, introduction to object oriented databases

Unit – II

Database Design and ER diagrams: Beyond ER Design; Entities, Attributes and Entity sets; Relationships and Relationship sets; Additional features of ER Model ;Conceptual Design with the ER Model ;Introduction to the Relational Model; Integrity Constraint Over relations ; Enforcing Integrity constraints; ; Logical database Design :ER to Relational. Introduction to Views: Destroying/Altering tables and Views. Relational Algebra – Selection and projection set operations – renaming – Joins – Division

Unit – III

SQL: Queries,Constraints,Triggers:Overview ;The Form of a Basic SQL Query ; Nested Queries:, Correlated Nested Queries, Set-Comparison Operators; Aggregative Operators ;

NULL values : Comparison using Null values; Logical connectives AND, OR, and NOT, Impact on SQL Constructs, Outer Joins ,Disallowing NULL values, Triggers and Active Data bases.

Unit – IV

Schema refinement and Normal forms: Problems Caused by Redundancy, Decompositions, Problem related to Decomposition; Functional Dependencies; Reasoning about FDS ; FIRST, SECOND, THIRD Normal Forms, BCNF ;Properties of Decompositions: Lossless join Decomposition, Dependency preserving Decomposition.

Transaction Concept; Transaction State; Implementation of Atomicity and Durability; Concurrent Executions ;Serializability; Recoverability; Lock –Based Protocols :Locks, Granting of locks,2PL,implementation of locking ; Timestamp Based Protocols.

Unit – V

Recovery System: Log – Based Recovery; Shadow Paging; Recovery with Concurrent Transactions;

File Organization and Indexing: Cluster Indexes, Primary and Secondary Indexes; Index Data Structures: Hash Based Indexing, Tree based Indexing; Comparison of File Organizations; B+ Trees Index

Text Books

1. Raghurama Krishnan, Johannes Gehrke: Database Management Systems.TATAMcGrawHill ,3rd Edition -2003
2. Silberschatz, Korth :Database System Concepts. McGraw hill,5th Edition – 2006.

Reference Books

1. Peter Rob, Carlos Coronel: DatabaseSystemsdesign Implementation and Management. Cengage Learning, 7th Edition
2. Elmasri, Navrate:Fundamentals of Database Systems. Pearson Education,6th Edition-2010
3. C.J. Date: Introduction to Database Systems. Pearson Education,4th Edition-2005.

Web links

1. https://www.coursera.org/course/db
2. **https://www.geeksforgeeks.org/dbms/dbms/**

AR26-MCA **I Year II Semester**

Computer Networks

Subject Code: 26MCA1007

L	T	P	C
3	0	0	3

Course Objectives:

- To Explain Data Communications System and its components, different types of network topologies and protocols.
- Apply knowledge of different techniques of error detection and correction to detect
- Solve error bit during data transmission.
- Analyze main protocols such as TCP, UDP.
- Analyze main protocols such as HTTP, FTP, SMTP.

Course Outcomes:

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

1. Identify and enumerate different types of network topologies, protocols and the layers of the OSI and TCP/IP models and explain the functions of each layer.
2. Explain the protocols of Data Link Layer and illustrate how a network can detect and correct transmission errors.
3. Classify, compare and analyze the major routing algorithms and distinguish congestion control algorithms and understand how a packet is routed over the internet.
4. Describe how TCP and UDP function, its uses and summarize the differences between them.
5. Analyze connection oriented and connectionless service.

Unit-I

Introduction: Data Communication, Components, Data Representation, Data Flow; Networks: Physical Structures, Categories of Network, Interconnection of Networks; The OSI models: Layered architecture, peer-to-peer process, Encapsulation, Layers in OSI model, TCP/IP protocol suite

Unit-II

Data Link Layer: Design Issues, Services Provided to Network Layer, Framing, Error Control and Flow Control, Elementary Data Link Protocols, Sliding Window Protocols.

Unit-III

The Network Layer: Network Layer Design Issues-Services: Provided to Transport Layer, Implementation of Connection Less Service, Implementation of Connection Oriented Service, Routing Algorithms: Shortest Path Routing, Distance Vector Routing, Link State Routing, The Network Layer in Internet-The IP Protocol, IP Address-IPV4, IPV6.

Unit-IV

The Transport Layer: Connectionless verses Connection Oriented Services, UDP: Well-known Ports for UDP, User Datagram, Checksum, UDP Operations and Uses of UDP; TCP: TCP Services, TCP Features, TCP Segment, a TCP Connection- Connection Establishment, Data Transmission, Connection Termination.

Unit-V

The Application Layer: DNS- Domain Name System- The DNS Name Space, Resource Records, Name Servers, Electronic Mail- Architecture and Services, The User Agent, Message Format, Message Transfer, Final Delivery, The World Wide Web- Architectural Overview, Hyper Text Transfer Protocol (HTTP)

Text Books:

1. Computer Networks: Andrew S Tanenbaum David J. Wetherall, 5/e, Pearson

2. Data communications and networking: Behrouz Forouzan, 5/e, McGraw Hill

3. Data and computer Communications, William Stallings, PEARSON,10/e

Reference Books:

1. An Engineering approach to computer Networking, S Keshav, 2nd Edition, Pearson Education.

2. Computer Networking a Top-Down approach featuring the internet, J.F. Kurose, K.W. Ross, 2nd Edition, Pearson Education.

Reference Links:

https://nptel.ac.in/courses/106105081

AR26-MCA **I Year II Semester**

OBJECT ORIENTED PROGRAMMING with JAVA

Subject Code: 26MCA1008

L	T	P	C
3	0	0	3

Course Objective

- To get the knowledge of basic concepts and techniques which form theobject oriented programming paradigm
- To understandtheconcepts of classes and objects and apply object-oriented programming principles to design software solutions.
- To apply inheritance and interfaces in Java for achieving code reusability, abstraction in object-oriented programming.
- To demonstrate the use of packages and working with Exception Handling.
- To learn and implement threads for concurrent programming and applets for developing interactive Java-based applications.

Course Outcomes

By the end of this course the student will be able to

1. Knowledge of the structure and model of the Java programming language
2. Explain the concept of class and objects with access control to represent real worldentities
3. Demonstrate the implementation of inheritance by using extends and implementskeywords
4. Illustrate different techniques on creating and accessing packages like fully qualifiedname and
import statement
5. Understand the impact of exception handling to avoid abnormal termination of program using
checked and unchecked exceptions and Use multithreading concepts todevelop inter process communication and also design applet Programs.

UNIT - I:

Introduction To Java: Evolution of Java, Java Buzzwords, The Java Virtual Machine, Anoverview of Java- Simple Java Program, Naming Conventions in Java, Data types, Variables,Expressions, Automatic type Conversion, Operators, Control Statements , Arrays, Strings.

UNIT - II:

Classes & Objects: Class fundamentals, Declaring Objects, Initializing the instance variables,Access Control, Constructors, Methods in Java, constructors, StaticMethods, Recursion, final keyword, this keyword, garbage collection, finalize() method

UNIT - III:

Inheritance: Inheritance Basics, Types of Inheritance, The Keyword „super", Final withinheritance. Polymorphism: Method Overloading, Method Overriding, Abstract Classes and methods.

INTERFACES: Interface, Multiple Inheritance using Interface, Abstract Classes vs. Interfaces

UNIT - IV:

Packages: Packages, Different Types of Packages, Access Protection, Importing Packages.

Exception Handling: Exception-handling fundamentals, throw Clause, throws Clause. Types ofExceptions: Built-in Exception, User Defined Exception.

UNIT - V:

Threads: Java Thread Model, Main Thread, Creating a Thread and Running it, terminating theThread, Creating Multiple Threads, Thread Synchronization, and Thread Priorities.

Applets: Applet Basics, Applet Life Cycle, A Simple Applet, HTML applet tag, AppletParameters.

Text Books:

1. Herbert Schildt, "Java The complete reference", 12thEdition, McGraw-Hill, 2022.

2. E.Balaguruswamy, "Programming with Java A Primer", 5th Edition, Tata McGraw-Hill, 2017.

Reference Books

1. Timothy budd, "An introduction to object-oriented programming", 3rdEdition,Pearson Education, 2009.

2. Y. Daniel Liang, "Introduction to Java programming", 9thEdition, Pearson education,2015

Reference Links

1. https://en.wikibooks.org/wiki/Java_Programming

2. https://nptel.ac.in/courses/106105191

AR26-MCA **I Year II Semester**

Design and Analysis of Algorithms

SubjectCode: 26MCA1009

L	T	P	C
3	0	0	3

Course Objectives:

- To analyze and express the efficiency of algorithms in terms of time and space complexity.
- Apply suitable algorithmic strategies to solve computational problem.
- Identify problems that require techniques such as greedy methods and dynamic programming.
- Compare different algorithms and select the most efficient solution.
- Understand the theoretical limits of algorithmic solutions, including lower bounds and NP-completeness.

Course Outcomes:

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

1. Analyze and evaluate the efficiency of algorithms using various performance metrics
2. Able to analyze and implement divide-and-conquer and greedy algorithms to solve complex problems.
3. Apply dynamic programming techniques to solve complex optimization problems s
4. Able to apply the backtracking method to solve complex combinatorial problems
5. Demonstrate an understanding of NP- Completeness theory and lower bound theory

UNIT – I:

Introduction and Complexity Analysis

Definition and characteristics of algorithms -Algorithm design issues-Performance analysis of algorithms-Time and space complexity-Asymptotic notations: Big-O, Ω, Θ – Best, average, and worst-case analysis.

UNIT – II:

Divide and Conquer and Greedy Techniques

Divide and conquer strategy-Recurrence relations and solving recurrences-Binary search -Merge sort-Quick sort-Greedy method and its characteristics-Activity selection problem-Knapsack problem-Huffman coding-Minimum spanning trees: Prim's and Kruskal's algorithms.

UNIT – III:

Dynamic Programming

Principles of dynamic Programming-Optimal substructure and overlapping sub problems Difference between greedy and dynamic programming-Matrix chain multiplication,Optimal binary search tree-0/1 Knapsack Problem-All pairs shortest path (Floyd–Warshall algorithm)

UNIT – IV:

Backtracking, Branch and Bound, and Graph Algorithms

Backtracking concept and state space tree-N-Queens problem-Sum of subsets- Graph coloring-Branch and bound technique-FIFO and LC branch and bound-Traveling Salesperson Problem-Graph traversal algorithms: BFS and DFS-Topological sorting

UNIT – V:

NP, NP-hard, and NP-complete classesReducibility-Examples of NP-complete problems

Text Books:

1. Fundamentals of Computer Algorithms, Ellis Horowitz, SatrajSahni and Rajasekharam, UniversitiesPress,2/e

2. The Algorithm Design Manual, 2nd edition, Steven S. Skiena, Springer

3. Introduction to Algorithms, second edition, T.H.Cormen, C.E.Leiserson, R.L.Rivestand C.Stein, PHI Pvt.Ltd

4. The Design and Analysis of Computer Applications low price edition, aho,Hopcroft,ullman

Reference Books:

1. Introduction to the Design and Analysis of Algorithms, Anany Levitin, PEA,3/e

2. Design and Analysis of Algorithms, Pearson Education, Parag Himanshu Dave, HimansuBalachandraDave,2/e

3. Introduction to Design and Analysis of Algorithms A strategic approach, R.C.T. Lee, S.S.Tseng, R.C.Chang and T.Tsai, McGrawHill.2005

4. Design and Analysis of algorithms, Pearson education, Aho, UllmanandHopcrof t,1/e

Reference Links:

https://nptel.ac.in/courses/106101059

AR26-MCA **I Year II Semester**

Python Programming

SubjectCode: 26MCA1010

L	T	P	C
2	0	0	2

Course Objectives:

- To equip students with the skills to use Python for problem-solving,
- Understand fundamental programming concepts,
- Apply these skills to real-world applications.
- To understand and apply object-oriented programming concepts in Python
- To explore Python libraries and their applications for efficient problem-solving, data analysis, and real-world software development.

Course Outcomes:

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

- Examine Python syntax and semantics and be fluent in the use of Python flow control and
  functions.
- Demonstrate proficiency in handling Strings and File Systems.
- Create, run and manipulate Python Programs using core data structures likeLists, Dictionaries
- Implement file handling functions and user defined functions inpython.
- Interpret the concepts of Object-Oriented Programming as used in Python andRegular Expressions.

UNIT - I:

Introduction to Python:

History, Features, Installing Python, Running Python, Python Keywords,Identifiers,Variables, Comments, Indentation, Operators, Built-in data types,Control Structures: Conditional Statements, Loops, Nested Loops.

UNIT - II:

Python Data Structures, Strings, File Handling:

Python Data Structures:Lists, Tuples, Sets, Dictionary

Strings- String Operations, Traversing a String, String handling Functions, File Handling

UNIT - III:

Functions and Modules: Functions: Functions, Built-in Functions, User Defined Functions, recursive functions, Scope of a Variable,Defining Functions-Function arguments: positional, keyword, default, variable length-Lambda Functions-, Input and output functions, modules and packages-import statements

UNIT - IV:

Object Oriented Programming: Object Oriented Programming: Class and object Constructor and destructor-Inheritance-Polymorphism-Encapsulation-Abstraction

Exception Handling- Exception in python, Types of Exception, User-defined Exceptions.

Getting Input, Importing MySQL for Python

Database Connectivity:Database connectivity (SQLite / MySQL).

UNIT - V:

Python Libraries & Applications:

Python Libraries: NumPy-Pandas-Matplotlib-SciPy- Seaborn-Applications of Python:Data Science

Machine Learning-Web Development-Automation.

Advanced Libraries: Plotly- Tensor Flow- PyTorch- Keras- NLTK- spaCy- OpenCV

Textbooks:

1. Wesley J .C hun "Core Python Applications Programming", 3rd Edition, 2012, Prentice Hall.

2. Brian jones, David Beazley “Python Cookbook”, 3rd Edition.

Reference Books:

1. Mark Lutz "Programming Python, 4th Edition" O'Reilly Media.

2. Think Python, Allen Downey, Green Tea Press

Reference Links:

1. nptel.ac.in/courses/106106182
2. https://pythonprogramminglanguage.com

AR26-MCA **I Year II Semester**

No SQL Databases

(Elective-1)

Subject Code: 26MCA1E11

L	T	P	C
3	0	0	3

Course Objectives:

- To understand the core concepts of NoSQL databases.
- To learn about Aggregate data model in NoSQL databases.
- To understand and apply data distribution models such as sharding, replication and consistency mechanisms in NoSQL databases
- To understand the design and implementation of key-value stores and user profile management using NoSQL databases
- To understand the design and use of document-oriented databases in NoSQL environments.

Course Outcomes:

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

1. Knowledge about NoSQL
2. Discuss about Aggregate Data Models
3. Explain about Master-Slave Replication, Peer-to-Peer Replication
4. Make use of Complex Transactions Spanning Different Operations
5. Understand about Document Databases

Unit-I

Why NoSQL: The Value of Relational Databases, Impedance Mismatch, Application andIntegration Databases, Attack of the Clusters, The Emergence of NoSQL

Unit-II

Aggregate Data Models: Aggregates, Column-Family Stores, Summarizing Aggregate-Oriented Databases

More Details on Data Models: Relationships, Graph Databases, Schema less Databases,Materialized Views, Modelling for Data Access

Unit-III

Distribution Models: Single Server, Sharding, Master-Slave Replication, Peer-to-Peer Replication, Combining Sharding and Replication

Consistency: Update Consistency, Read Consistency, Relaxing Consistency, Relaxing Durability, Quorums

Unit-IV

What Is a Key-Value Store, Key-Value Store Features, Consistency, Transactions, QueryFeatures, Structure of Data, Scaling, Suitable Use Cases, Storing Session Information, UserProfiles, Preference, Shopping Cart Data, When Not to Use, Relationships among Data, Multioperation Transactions, Query by Data, Operations by Sets.

Unit-V

Document Databases, What Is a Document Database, Features, Consistency, Transactions, Availability, Query Features, Scaling, Suitable Use Cases, Event Logging, Content Management Systems, Blogging Platforms, Web Analytics or Real-Time Analytics,Ecommerce Applications, When Not to Use

Text Books:

1. Pramod J. Sadalage, Martin Fowler. NoSQL Distilled, Addison Wesley 2013.

2. Shashank Tiwari, Professional NoSQL, Wrox Press, Wiley, 2011.

Reference Books:

1) Luc Perkins, Eric Redmond, Jim R. Wilson. ThePragmatic Bookshelf, 2018

2) Gaurav Vaish, Getting Started with NoSQL, Packt Publishing, 2013.

Reference Links:

https://www.udemy.com/course/introduction-to-nosql-databases/

MOBILE APPLICATION DEVELOPMENT

(Elective-1)

L	T	P	C
3	0	0	3

Subject Code: 26MCA1E12

Course Objectives:

- To introduce characteristics of mobile applications. Application models of mobile application frameworks.
- Managing application data and User-interface design for mobile applications.
- Integrating networking, the OS and hardware into mobile-applications.
- To demonstrate their skills of using Android software development tools.
- To demonstrate their ability to deploy software to mobile devices.

Course Outcomes:

Upon completion of the course students should be able to:

1. Install and configure Android application development tools
2. Design and develop user Interfaces for the Android platform
3. Save state information across important operating system events
4. Apply Java programming concepts to Android application development
5. Understanding mobile architecture, mastering UI design, managing local/cloud data, using device sensors and implementing secure, user-friendly functionalities.

UNIT I

Introduction to mobile devices: Introduction to Mobile Computing, Introduction to Android Development Environment, Mobile devices vs. desktop devices, ARM and Intel architectures, Screen resolution, Touch interfaces, Application deployment, App Store, Google Play.

UNIT II

Android User Interface: Measurements – Device and pixel density independent measuring units User Interface (UI) Components – Editable and non-editable Text Views, Buttons, Radio and Toggle Buttons, Checkboxes, Spinners.

UNIT III

Back Ground Running Process, Networking and Telephony Services: Services: Introduction to services – local service, remote service and binding the service, the communication between service and activity, Intent Service.

UNIT IV

Android: Introduction – Establishing the development environment – Android architecture – Activities and views – Interacting with UI – Persisting data using SQLite – Packaging and deployment – Interaction with server side applications – Using Google Maps, GPS and Wi-Fi – Integration with social media applications.

UNIT V

Advanced Topics: Power Management: Wake locks and assertions, Low-level OS support, Writing power-smart applications. Augmented Reality via GPS and other sensors: GPS, Accelerometer, Camera.

Text Books:

1) Bill Phillips, Chris Stewart, Brian Hardy, and Kristin Marsicano, Android Programming: The Big Nerd Ranch Guide, Big Nerd Ranch LLC, 2nd edition, 2015.

2) Valentino Lee, Heather Schneider, and Robbie Schell, Mobile Applications: Architecture, Design and Development, Prentice Hall, 2004.

3) ProfessionalAndroid4ApplicationDevelopment, Roomier, Wiley India, (Wrox), 2012

Reference Books:

1) Beginning Android 4 Application Development, Wei-Meng Lee, Wiley India (Wrox), 2013

2) Tomasz Nurkiewicz and Ben Christensen, Reactive Programming with RxJava, O"Reilly Media, 2016.

WEB LINKS:

https://www.udemy.com/courses/development/mobile-apps/?srsltid=AfmBOopoa8dYQbCVpwmAjAMRR1EcE6MK30c SvTN9ATWMpYydSRC7Z_y

https://www.mygreatlearning.com/mobile-app-development/free-courses

Artificial Intelligence

(Elective-1)

L	T	P	C
3	0	0	3

Subject Code: 26MCA1E13

Course Objective:

- To use a basic AI programming language to write simple to medium-level programs and understand existing AI programs.
- Learn the basics of knowledge representation, blind and heuristic search methods, and important AI concepts such as minimax and resolution used for problem solving.
- To understand advanced topics in artificial intelligence to support further learning and practical use.
- To understand and apply knowledge representation techniques for organizing and modeling information in intelligent systems.
- To understand the principles of expert systems and fuzzy logic and apply them for building intelligent systems.

Course Outcomes:

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

1. Analyze and classify problem domains to determine their suitability for artificial intelligence

Techniques and select appropriate AI methods for solution development.

2. Formulate and model problems using the concepts, languages, and frameworks of diverse AI

Paradigms.

3. Develop and implement fundamental AI algorithms to solve well-defined computational

problems.

4. Design, conduct, and analyze empirical evaluations to compare the performance of AI

algorithms under different problem formulations.

5. Interpret and justify conclusions drawn from experimental results using quantitative and

Qualitative performance metrics.

UNIT - 1:

Introduction: History, intelligent systems, introduction to intelligent agents, foundations of AI, applications, tic- tac-toe game playing, development of AI languages, current trends.

UNIT - 2:

Solving Problems by Searching- Search Strategies: State-space problem,Problem Solving by Intelligent search: BFS, DFS, Iterative Deepening Search, Hill Climbing,Simulated Annealing, heuristic Search: A*, AO* Algorithms.

UNIT - 3:

Logic concepts: Introduction, propositional calculus,propositional logic, natural deduction system, axiomatic system, semantic tableau system in propositional logic, resolution refutation inpropositional logic, predicate logic.

UNIT - 4:

Knowledge Representation:

Introduction to knowledge representation, approaches to knowledge representation, semantic networks and extended semantic networks, frame-based knowledge representation, followed by advanced techniques such as conceptual dependency theory, script structures, and case grammars.

UNIT - 5:

Expert Systems and Fuzzy Logic:

Introduction to expert systems, comparison with traditional systems, and phases in building expert systems, An introduction to fuzzy sets and fuzzy logic, fuzzy set operations, membership functions, multi-valued logic, linguistic variables, fuzzy propositions, inference rules, and fuzzy systems.

Text Books

1. Artificial Intelligence- Saroj Kaushik, CENGAGE Learning.

2. Artificial intelligence, A modern Approach, 2nded, Stuart Russel, Peter Norvig, PEA.

Reference Books

1. Artificial Intelligence- Deepak Khemani, TMH,2013.

2. Introduction to Artificial Intelligence, Patterson, PHI.

3. Artificial intelligence, structures and Strategies for Complex problem solving, George FLugar, 5th ed, PEA

Reference Link:

https://nptel.ac.in/courses/106102220

AR26-MCA **I Year II Semester**

Computer Vision and Image Processing
(Elective-1)

Subject Code:26MCA1E14

L	T	P	C
3	0	0	3

Course Objectives:

- To comprehend the relation between human visual system and machine perception processing of digital images.
- To provide a detailed approach towards image processing applications like enhancement, segmentation, and compression.
- To understand and apply the Hough Transform technique for detecting geometric shapes in digital images.
- To understand and apply computer vision techniques for solving real-world problems such as image analysis, object detection, and pattern recognition.

Course Outcomes:

Computer Applications:

1. Demonstrate methods for 3D vision.
2. Explain 3D object recognition, adjustments, and alignments.
3. Apply Face detection techniques.
4. Plan In-Vehicle vision system- Identify road signs, locating pedestrians.
5. Interpret image segmentation and representation techniques.

UNIT I

Image Formation and Coordinate Transformations Camera Matrix, Motion/Stereo Pin-hole model, Human eye / cognitive aspects of color / 3D space; illumination; Sampling and Quantization Coordinate transformations and camera parameters.

UNITII

Image Processing - Noise Removal, Blurring, Edge Detection: Canny / Gaussian/ Gabor/Texture Edges/ Curvature / Corner Detection.

UNIT III

HOUGH TRANSFORM - Line detection – Hough Transform (HT) for line detection – foot-of-normal method – line localization – line fitting – RANSAC for straight line detection – HT based circular object detection – accurate centre location – speed problem – ellipse detection – Case study: Human Iris location – hole detection – generalized Hough Transform (GHT).

UNIT IV

3D VISION AND MOTION (10Lectures) Methods for 3D vision – projection schemes – shape from shading – photometric stereo – shape from texture – shape from focus – active range finding – surface representations – point-based representation – volumetric representations – 3D object recognition – 3D reconstruction – introduction to motion – triangulation – bundle adjustment.

UNITV

Applications: Photo album – Face detection – Face recognition – Eigen faces – Active appearance and 3D shape models of faces Application: Surveillance – foreground-background separation – Particle filters.

Text Books

1. D. L. Baggio et al., ―Mastering OpenCV with Practical Computer Vision Projects‖, Packt Publishing, 2012.

2. E. R. Davies, ―Computer & Machine Vision‖, Fourth Edition, Academic Press, 2012.

3. CosimoDistante, Sebastiano Battiato., ―Image Processing and Vision Engineering, 5th International Conference, IMPROVE 2025

ReferenceBooks:

1. Jan Erik Solem, ―Programming Computer Vision with Python: Tools and algorithms for analysing images‖, O'Reilly Media, 2012.
2. Mark Nixon and Alberto S. Aquado, ―Feature Extraction & Image Processing for Computer Vision‖, Third Edition, Academic Press, 2012.
3. R. Szeliski, ―Computer Vision: Algorithms and Applications‖, Springer 2011.
4. Simon J. D. Prince, ―Computer Vision: Models, Learning, and Inference‖, Cambridge University Press, 2012.

Web Links:

https://www.geeksforgeeks.org/computer-vision/computer-vision/

https://www.geeksforgeeks.org/electronics-engineering/what-is-image-processing/

AR26-MCA **I Year II Semester**

Database Management Systems Lab

Subject Code: 26MCA1104

L	T	P	C
0	0	3	1.5

Course Objective

- To learn defining Logical Database schema, Query writing to retrieve required information from single/multiple tables, Creation and manipulation of views.
- To Implement Operations on relations (Tables) using PL/SQL,Writing Triggers for implementing automatic operations and constraints,
- To write Cursors, Functions and Procedures for various tasks on tables, Exception handling and Packages.

Course Outcomes

After completion of this course, the Students will be able to:

1. Create relational database
2. Manipulate data using SQL
3. Compose Queries to retrieve required information from Database
4. Develop programs using Triggers and Cursors
5. Design Procedures, Functions and Packages for required Database tasks.

List of Experiments

1. Execute DDL and DML and commands employee table.
2. Execute group functions on employee table.
3. Createan employee table for various relations in SQL with necessary integrity constraints, keys, data types. Verify messages by violating the constraints
4. Perform various join operations like Equi and non-equi, outer join, self-join onany two tables and show the results.
5. Implement Nested and correlated nested queries sailors,reserves and boats table
6. Write a PL/SQL programs using iteration statements and conditional statements
7. Write a PL/SQL program to handle any one built-in and user defined exceptions On employee table
8. Write a PL/SQL program using Cursors to display the records of employee table
9. Write a PL/SQL program using triggers to save deleted rows from employee table
10. Write a PL/SQL program to implement a function to find the maximum of 2 numbers
11. Write a PL/SQL program to create a Package and use the package elements later.

Reference Books

SQL, PL/SQL the Programming Language of Oracle by Ivan Bayross, BPB Publications, 4th Edition.

AR26-MCA **I Year II Semester**

Computer Networks Lab

L	T	P	C
0	0	3	1.5

Subject Code: 26MCA1105

Course Objective

- To understand the fundamental concepts of computer networking and the OSI reference model. To become familiar with the basic taxonomy and terminology used in the field of computer networking.
- To learn and understand advanced networking concepts, preparing them for higher-level courses in computer networking.
- To develop knowledge and practical understanding in specific networking areas, including the design and maintenance of individual computer networks.

Course Outcomes

After completion of this course, the students will be able to:

1. Identify various data transmission techniques and modes.
2. Explain data link layer protocols, multiple access protocols, and IEEE 802 LAN standards.
3. Describe routing, congestion control, routing algorithms, and IPv4 addressing.
4. Discuss transport layer functions and protocols.
5. Outline basic network security and application layer protocols such as FTP, HTTP, Telnet, and DNS.

List of Experiments

1. Implement different data link layer framing methods.
2. Implement error detection techniques
3. Implementation of routing algorithms
4. Implementation of congestion control algorithms.
5. Implement the following forms of IPC.
   a. Pipes b. FIFO
6. Use semaphores to prevent race conditions when two processes run at the same time.
7. Design TCP Client and server application
8. Design UDP Client and server application
9. Apply basic networking commands to test and troubleshoot network connectivity through a program.
10. Create a basic protocol header using structures and bit fields.
11. Design an experiment using CISCO packet tracer

Reference Books

1. Computer Networks –ASystemApproach, Peterson, BruceDavie,2/e, Harcourt Asia
2. Compute communications and networking technologies, Gallo, Hancock, Cengage
3. An Engineering approach to compute networking, Kesha, Pearson

JAVA Programming Lab

Subject Code: 26MCA1106

L	T	P	C
0	0	3	1.5

Course Objective:

- To develop skills to design and analyze the applications with respect to java programming
- To strengthen the ability to identify and apply the suitable object oriented concept for the given real world problem

Course Outcomes:

By the end of this course the student will be able to

- Able to write, compile and execute simple java programs
- Explain the concept of class and objects with access control to represent real world entities
- Use overloading methodology on methods and constructors to develop application programs
- Describe the concept of interface and abstract classes to define generic classes
- Able to create user defined packages and handle exceptions at run time and Apply Threading
- concept based on application requirement and design Applet programming

LIST OF EXPERIMENTS

1. A) Write a java program that displays welcome to follow by user name.
   Accept username from the user.

B) Write a java program that prompts the user for an integer and then prints out all the prime numbers up to that integer.

2. A) Write a java program to create a class Rectangle. The class has attributes Length and Width.

It should have methods that calculate Area and Perimeter of the Rectangle. It should have read

Attributes () method to read Length and Width from the user.

B) Write a java program for Fibonacci sequence and defined as: The first two values in the sequence are1 and 1. Every subsequent value is the sum of the two values preceding it.

3. A) Write a java program that uses both Recursive and Non-Recursive functions to find the factorial of a given number.

B) Write a java program that checks whether the given string is Palindrome or not.

Ex:MALAYALAM is a Palindrome.

4. A) Write a java program to illustrate method overloading and method overriding.

B) Writ a java program that illustrates how java achieved Run Time Polymorphism.

5. A) Write a java program to demonstrate the use of subclass.

B) Write a java program for abstract class to find areas of different shapes

6. Write a Java program to implement the concept of importing classes from user defined package and creating packages.

7. Write a java program to implement the concept of Exception Handling by using predefined and user defined exceptions.

8. Write a java program to implement the concept of Threading by Extending Thread class and by Implementing Runnable Interface.

9. Write a program using Applet to display a message in the Applet and for configuring Applets by passing parameters.

10. Write a java program to implement thread priorities

Python Programming Lab

L	T	P	C
0	0	3	1.5

SubjectCode: 26MCA1107

Course Objectives:

- Enable students with a comprehensive understanding of Python programming language, including its history, features, and fundamental concepts
- Enabling them to write Python programs proficiently.

Course Outcomes:

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

1. Understand basic Python syntax, printing, and data types.
2. Work with lists, tuples, and dictionaries to store and manage data.
3. Write functions using lambda, recursion, and variable arguments.
4. Handle errors and perform file operations in Python.
5. Connect Python programs with a database and execute SQL queries.

Exercise 1

Ex 1: Declare and use identifiers belonging to strings, integers, floats, and booleans.

Ex 2: Accept user input (name, age, height, student status) and display each value with its type using type().

Exercise 2

Ex 3: Perform operations like upper(), find(), replace() on strings.

Ex 4: Write a program to reverse the string, count vowels and words.

Ex 5: Write a program for slicing, sorting, and list comprehension.

Exercise 3

Ex 6: Program to apply list methods: append(), extend(), insert(), remove(), pop(), sort().

Ex 7: Create tuples to store student (name, age, course) data and perform

a. Accessing elements using indexing and slicing.

b. Demonstrate immutability by attempting to modify a tuple.

c. Create and navigate nested tuples.

Ex 8: Create a dictionary with student roll numbers as keys and names/marks as values.

a. Accessing, adding, updating, and deleting key-value pairs.

b. Iterating through keys, values, and items.

Exercise 4

Ex 9: Write a program to demonstrate variable length arguments.

Ex 10: Write a program to illustrate lambda and recursive functions.

Ex 11: Write a program to demonstrate Globals(), Locals(), and Reload() functions.

Exercise 5

Ex 12: Demonstrate exception handling and assertions in Python.

Ex 13: Write a Python program to copy the contents of one file into another in reverse order.

Ex 14: Write a program to connect to the database and retrieve the required information using SQL commands.

Text Books:

1. Wesley J .C hun "Core Python Applications Programming", 3rd Edition, 2012, Prentice Hall.

2. Brian jones, David Beazley “Python Cook book ”, 3rd Edition.