

**PITHAPUR RAJAH'S GOVT. COLLEGE
(A), KAKINADA**



**DEPARTMENT OF
COMPUTER APPLICATIONS**

INFORMATION TECHNOLOGY

**BOARD OF STUDIES
2026-2027**

PROCEEDINGS OF THE PRINCIPAL (FAC), PITHAPUR RAJAH's GOVT. COLLEGE [A], KAKINADA
Present: Dr. Kandula Anjaneyulu, M.A, Ph.D.

Rc.No.4/A.C/BOS- UG & PG /2026-27

Dt.17 June 2026

Sub: P.R. Government College (Autonomous), Kakinada – Academic Cell – Conduct of Board of Studies (BoS) Meetings for UG and PG Programmes for the Academic Year 2026–27 – Communication of Guidelines – Regarding.

ORDER:

The autonomous colleges, in alignment with their vision, mission, stated objectives, and core values, are mandated to design and develop their own outcome-based curricula. This must be done with due consideration for societal, local, and global industry requirements, employability, and the development of industry-ready and transferable skills. Accordingly, every programme shall prescribe Course Outcomes (COs), Programme Outcomes (POs), and Programme Specific Outcomes (PSOs) along with a suitable learning outcome assessment management system, supported by a robust and transparent evaluation mechanism to measure attainment levels among students.

Further, the A.P. State Council of Higher Education (APSCHE) has introduced a revised curricular framework effective from the Academic Year 2025-26, incorporating Skill Enhancement Courses, Multi-Disciplinary courses, the Indian Knowledge System and a revised credit structure.

Our institution, from the Academic Year 2022–23 onwards, has defined a renewed vision and mission along with updated objectives and core values, necessitating the design and reorientation of its academic and research administration in line with these directives.

In light of the above responsibilities prescribed by the institution's vision and mission, NEP–2020, NAAC, NIRF, and the APSCHE's revised and new UG and P.G. curricular framework, it is imperative to customize, design, and re-orient our academic and research activities to meet the expectations of students, industries, and government stakeholders.

Accordingly, the Chairpersons of the U.G and P.G Boards of Studies (BoS) of various departments are hereby requested to make necessary arrangements to convene their BoS meetings from **25 June to 30 June 2026**.

The Chairpersons are further instructed to:

1. Prepare the curricula and extracurricular activities for the Academic Year 2026–27 in line with the institution's vision, mission, NEP–2020, and NIRF norms.
2. Devise an appropriate evaluation system to ensure effective learning outcomes and holistic student development.
3. Ensure that the curriculum design includes a **20% revision** of the syllabus each year without deviating from the APSCHE prescribed syllabus.

4. If the syllabus is not prescribed by APSCHE/Affiliating University, then the syllabus is to be framed by the BOS committee concerned with duly following the mandate prescribed above.
5. Engage stakeholders viz employers, parents, and alumni, to obtain feedback on the existing curricula and to invite suggestions for improvements.
6. Invite the University nominee, subject experts, industry representatives, student representatives, and parent representatives well in advance. The meeting notice shall clearly specify the date, venue, and agenda, and a soft copy of the agenda and relevant documents shall be circulated for their perusal.
7. Ensure that the subject experts invited preferably hold a Doctorate with at least 10 years of teaching experience and have relevant expertise in designing industry-related, market- and job-oriented curricula.
8. Facilitate thorough deliberations on curriculum design, evaluation methods, incorporation of research components, measures to enhance learning experiences, and optimal utilization of existing human, physical, and ICT resources.
9. Conduct all BoS meetings in offline mode. Online participation shall be permitted only under exceptional circumstances.
10. Prescribe benchmarking and quality initiatives in pedagogy and learning, including strategies for curriculum design and teaching-learning processes, in collaboration with the IQAC Coordinator, prior to the BoS meeting.
11. Ensure that a minimum student attendance of **75%** shall be required for eligibility to appear for I & II Mid-Term Examinations under the CIA component; this shall be formally approved in the BoS meeting.
12. Approve any new programmes to be introduced for the Academic Year 2026–27, the number and frequency of certificate courses, and SWAYAM MOOCs courses.
13. Submit the approved BOS copies in the prescribed format, in **quadruplicate (hard copies)** to the Academic Cell for onward submission to the IQAC, Examination Cell, and Library, within **three days** of the meeting and upload the soft copy in their respective department web pages in the college website.
14. Ensure strict alignment of all recommendations and curriculum changes with the institution's vision and mission.
15. Submit a request to receive advance funds from the Examination cell through Principal for conducting BoS meetings.

Following contents shall be presented in the BOS document in the order

1. Proceedings of the Principal pertaining to BOS
2. Composition of BOS
3. Vision and Mission of the department
4. Agenda: It shall include ATR on the previous BOS meeting first, resolutions, etc., later.
5. Table showing the Allocation of Credits in the following table for both theory and Practicals' in case of science subjects

S. No	Semester	Title of the Course (Paper)	Hrs./week	Max. Marks (SEE)	Marks in CIA	Credits
1	III	Physical Chemistry-1	3	50	50	4

6. Resolutions adopted in the meeting with detailed discussion that took place during the meeting.
7. Each BOS Chairman shall, immediately after syllabus, tabulate the changes made in the syllabus/ paper along with justification.
8. Attendance of Members present with signatures in the tabular form.
9. List of Examiners & Paper setters (Minimum 20 members and at least 02 members from other states)
10. Syllabus for each course (both theory & Practical in case of Science subjects) followed by model question papers (theory & practical) and allocation of CIA (50marks) for each course with structure.

CIA structure for Single Major system

- Out of 50 marks for CIA, 25 marks are allocated for Mid examinations. In each semester two mid examinations to be conducted and the average of the two will be considered.
- Mid examinations are to be conducted in offline mode at college level
- Mid examination to be conducted in offline mode in which the student should attempt **one essay** question for ten marks out of two questions, **three short** answer questions with five marks each out of five questions
- The remaining 25 marks for CIA are allocated as per the following structure.

Project-10M	Seminar- 5M	Assignment- 5M	Viva on theory- 3M	Clean & green and Attendance- 2M
-------------	-------------	----------------	--------------------	----------------------------------

BOS COMPOSITION
PROCEEDINGS OF THE PRINCIPAL (FAC), P.R. GOVERNMENT COLLEGE (A),
KAKINADA

Present: Dr. Kandula Anjaneyulu, M.A, Ph.D.
R.C.No.05/A.C/BOS Members Nomination/2026-27

Dated: 17.06.2026

SUB: P.R. Government College (A), Kakinada- UG Board of Studies (BOS)- Nomination of Members - Orders issued.

REF: Proc.RC.No.4/A.C/BOS/2026-27 dated:17 June 2026 of the Principal, Pithapur Rajah's Government College(A) Kakinada.

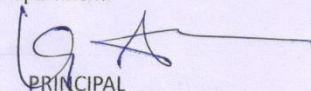
ORDER:

The Principal, P.R. Government College(A), Kakinada is pleased to constitute UG Boards of Studies in **B.Sc-Information Technology** for framing the syllabi in respective Subject for all semesters duly following the norms of the UGC Autonomous guidelines.

S. No	Name of the Person	Designation
1	Smt P.Jyothi	Chairman & Lecturer Incharge, Department.
2	Smt.N.NagaSubrahmanyeswari Lecturer In Computer Science ASD Govt degree College for Women(A) Kakinada	University Nominee
3	Dr K.V.Sobha Rani, Lecturer in Computer Applications, GDC Ramachandrapuram	Subject Expert-1
4	Dr P.Vamsi Krishna Raja Professor Computer Science and Engineering, Pydah College of Engineering, Kakinada.	Subject Expert-2
5	P.Swamy Vandanam, M.D, BDPS Computers	Representative from Industry
6	Ms. N.Sravani Devi	Member
7	Ms. S.Devi Prasanna	Member
8	Ms. P.V.Maheswari	Member
9	Ms. K.Sravani Devi	Member
10	Mrs P.Sanghavi	Member
11	Mrs. K.Veera Veni	Member
13	G.S.N Tejaswi	Student Alumni Member
14	B.Santosh	Student Member
15	K.Bhagyasri	Student Member

The above members are requested to attend the BoS meeting on 01-07-2026 AN, and share their valuable reviews, and suggestions on the following functionaries.

- Prepare syllabi for the subject keeping in view the objectives of the college and interest of the stake holders for consideration and approval of the IQAC and Academic Council.
- Suggested methodologies for innovative teaching and evaluation techniques.
- Suggest the panel of Names to the academic council for appointment of Examiners.
- Coordinate research, teaching, extension and other activities in the Department.


PRINCIPAL

P. R. Government College(A), Kakinada

The Chairpersons of all Boards of Studies are hereby instructed to comply with these directives in letter and spirit to ensure the highest standards of academic and administrative excellence



PRINCIPAL
Pithapur Rajah's Government College(A)
Kakinada

Copy to:

1. Lecturers-in-Charge (BOS Chairmen) of all the departments
2. Academic Coordinator
3. IQAC coordinator
4. Controller of Examinations
5. Office

DEPARTMENT OF COMPUTER APPLICATIONS

Vision:

Department of Computer Applications Strives rigorously to provide quality education in both theoretical and applied foundations of computers and train the students to solve the real world problems effectively thus enhancing their potential for high quality careers.

Mission:

- To produce students with critical thinking and lifelong learning capabilities for applying their knowledge to uplift the living standards of the society.
- To equip students and faculty with excellent teaching-learning capabilities through advanced learning tools and technologies.
- To produce students with enriched skill set, professional behaviour, strong ethical values and leadership capabilities so as to work with commitment for the progress of the nation.

PSO's for BSc-Information Technology:

PSO 1: Students will improve their computer literacy, their basic understanding of operative systems and a working knowledge of software commonly used in academic and professional environments

PSO 2: Students will learn to organize information efficiently in the forms of outlines, charts , etc. using appropriate software.

PSO 3: Students will acquire skills for designing and delivering an effective presentation and developing the various IT skills to electronic databases.

PSO 4: Students understand the internal operations and architecture of computers.

.

PSO 5: Students will be able to design and develop any software application.

PITHAPUR RAJAH'S GOVERNMENT COLLEGE [AUTONOMOUS] KAKINADA
DEPARTMENT OF COMPUTER APPLICATIONS
BOARD OF STUDIES 2026-2027

The third meeting of Board of Studies **INFORMATION TECHNOLOGY** has been conducted in the Department of Computer Applications on 01st July, 2026 at 1 PM to discuss the following.

Agenda

1. To approve the curriculum, blue print and model paper for 1ST year B.Sc Information Technology Major Course under CBCS based as per the directions of the APSCHE for the admitted batch 2026 -27 (I & II Semesters) (Syllabus as per APSCHE)
2. To approve the curriculum, blue print and model paper for 2nd year B.Sc Information Technology Major Course under CBCS based under the directions of the APSCHE for the admitted batch 2025 -26(III & IV Semesters) To approve the curriculum, blue print and model paper for 3RD year B.Sc Information Technology, under CBCS based under the directions of the APSCHE for the admitted batch 2024 -25 (V & VI Semester)
3. To approve the Two Certificate Courses for Information Technology students were introduced in this academic year.
4. To approve the incorporation of additional inputs to various courses (where ever it is felt necessary) for enhancing students understanding over the concerned course and this shall not be considered for evaluation purpose.
5. To approve the Examination procedure for the courses for I, II, III years of B.Sc Information Technology (2026 – 25, 2025-24 & 2024-25 admitted batches).
6. To approve the Scheme of Valuation for Practical's
7. Every Student has to complete one MOOC's course per year. And the Student award two extra credits to who have registered and completed SWAYAM course successfully.
8. To award 2 credits for each first year Community service project (CSP) , and second year Internship between 1st and 2nd year and 2nd and 3rd year (two summer vacations) and 4 credits for Apprenticeship during 6th semester.
9. To implement pedagogical strategies to enrich teaching and learning process.
10. To approve the proposed departmental activities for 2026-27.
11. To approve the list of examiners and paper setters for the academic year 2026-27.
12. Mandatory attendance of 60% for I mid examinations, 75% of attendance for II mid examination and 75% attendance for SEE – theory and practical.
13. Introducing of New Courses of Study and the possibilities
14. Any other item with the permission of the chair.

CHAIRMAN
BOARD OF STUDIES

Resolutions taken:

The following resolutions are approved by university nominee and all the members of BOS After reviewing the existing titles and contents of class I,II,III,IV and V framed by APSHE, a the board come out with the following resolutions.

Resolution – I

It is resolved to adopt the Computer Science Major subjects for Information Technology course as the syllabus for Information Technology Course not provided by APSCHE following changes of course I,II of Information Technology as specified, from the academic year as, NEP 2020, the Major and minor policy system has come into effect.

It is resolved to approved the curriculum, blue print and model paper for 1st year B.Sc(Information Technology) Course under CBCS based as per the directions of the APSCHE for the admitted batch 2026 -27. (I Semester)

1st year : 1st sem – major(2 subjects)

1st year : 2nd sem – major(2 subjects)

Question Paper Model Pattern

Major

Section-I

Part-A : Given 3 questions

Part-B : Given 3 questions

Write any three questions, at-least one question from each part each question carries 10 marks

Section-II

Write any Four questions out of seven questions each question carries five marks.

Resolution – II

It is resolved to approve the syllabus for 2nd and 3rd year of B.Sc Information Technology as it is given by APSCHE as a part of this, from the academic year as, NEP 2020, the Major and minor policy system has come into effect. According to this

2nd year- IIIrd sem total subjects are: 3 -Major and 1-Minor

2nd year -IVth sem total subjects are: 3 -Major and 1-Minor.

3rd year- Vth sem total subjects are: 4 -Major and 2-Minor

Question Paper Model Pattern

Major Subject

Section-I

Part-A : Given 3 questions

Part-B : Given 3 questions

Write any three questions, at-least one question from each part each question carries 10 marks

Section-II

Write any Four questions out of seven questions each question carries five marks.

Resolution – III:

It is resolved to introduce certificate course with the Name **Indian Knowledge System in Natural Language Processing** (30 hours) as per the requirement, possibility and feasibility, another certificate course name is **Emerging and Advanced Technologies**.

Resolution – IV

It is resolved to approved the incorporation of additional inputs to various courses (where ever it is felt necessary) for enhancing students understanding over the concerned course and this shall not be considered for evaluation purpose.

Resolution – V

a) Each theory subject is evaluated for 100 Marks (I, II&III Years) out of which 50 Marks through semester end examination for I, II & III year, 50 marks for internal assessment.

b) The minimum pass mark for both internal and external examinations is 18 marks (36%), but as a whole student is subjected to get 40% marks (40 out of total 100 marks) to pass the subject. (I, II&III Years)

c) CIA structure for Single Major system

- Out of 50 marks for CIA, 25 marks are allocated for Mid examinations. In each semester two mid examinations to be conducted and the average of the two will be considered.
- I mid examination is to be conducted in offline mode at college level and II mid examination is to be conducted in online mode at department level.
- I mid examination to be conducted in offline mode in which **one essay** question for ten marks out of two questions, **two short** answer questions with five marks each out of four questions and five objective questions to be given for each paper.
- The remaining 25 marks for CIA are allocated as per the following structure.

Project 10M	viva on theory- 3M	assignment- 5M	Seminar- 5M	clean & green and Attendance- 2M
----------------	-----------------------	-------------------	-------------	-------------------------------------

d) Internal assessment for 50 Marks is as follows: (For Certificate Courses)

vii) Study Project : 20 Marks

viii) Student Seminar : 10 Marks

ix) Viva-voce : 10 Marks

x) Assignment : 10 Marks

Resolution – VI

it is resolved to set the Scheme of Valuation for Practical's

- Record - 10 Marks
- Viva voce - 10 Marks
- Test - 30 Marks
- Total - 50 marks

Resolution – VII

It is resolved to give extra credits for MOOCS courses, N.S.S., N.C.C., winners of zonal level sports and games competitions, participation in state level/ National level competitions, blood donations camps, environmental programs like extending services in facing the natural calamities etc.

Resolution – VIII

It is resolved to adopt Community Service Project for all the students at the end of Sem–II. Resolved to send all the students for Short-term Internship at the end of Sem –IV and it is Resolved to send all the final year students for on job training apprenticeship in connection with industries for off-site Project in the Semester VI with the industries in accordance with their interest of study.

Resolution – IX

It is resolved to implement pedagogical strategies to enrich teaching and learning process. It is Resolved to conduct International / National , Webinar / Seminar like Data Science, etc., It is Resolved to conduct extension lectures by the eminent persons.

Resolution – X:

It is resolved to approve the proposed departmental activities for 2026-27.

Resolution – XI:

It is resolved to approve the list of examiners and paper setters for the academic year 2026-27.

Resolution – XII

Streamlining of regularity in attendance. Resolved to make the eligibility to appear for 1st mid is 75% of attendance for the 2nd mid it would be 75% , for 75% of attendance for semester examination and 90% for practical examinations .Also it is resolved that the student should attend at least one internal exam to appear for the Semester end examination.

Resolution – XIII

It is Resolved to introduce new courses of study whenever necessary.

Resolution – XIV

It is Resolved to follow the admission criteria for the programmes offered by the department. It is resolved to arrange a field trip.

Department of Computer Applications

PANEL OF NAMES FOR APPOINTMENT OF EXAMINERS/PAPER SETTERS

2026-27

S.No	NAME & DESIGNATION	COLLEGE	EXPERIENCE	Address
1.	Smt S.Vaani Kumari	Govt. Degree College(W) (A) , Srikakulam e-mail: vanikumari.s@gmail.com Ph:9885766144	8 Years	C/o college
2.	Smt B.Lakshmi	SKR GDC, Gudur, Email: jlr.lakshmi@gmail.com Ph:7893204741	8 Years	C/o college
3.	Smt. J. Sharmila Rani	GDC, Gajapathinagaram, Vijayanagaram Dt, AP Email: sharmilarani.j@gmail.com Ph:8341548582	8 Years	C/o college
4.	Smt K.Adilakshmi	GDC Tiruvuru e-mail: computerdeptmpl@gmail.com Ph:9440027500	8 Years	C/o college
5.	Smt A. Madhavi	GC for (M)(A), Ananthapur Ph:9949672448	8 Years	C/o college
6.	Smt D Aruna Padma	GDC(W), VISAKHAPATNAM 9030615618	8 Years	C/o college
7.	Smt B. Durga Anuja	GDC for Women, Srikalahasti Ph no:9908236775	8 Years	C/o college
8.	Mr T. Narendra Babu	Dr YSR. GDC, Vedurukuppam, Ph: 7702917803	5 years	C/o college
9.	Mr B.Ravi	GDC Vijayanagaram Ph: 9494332331	5 years	C/o college
10.	D.Jyothi	CMR Institute of Technology, Hyderabad	9 years	C/o college
11.	B. Nidarshini	GDC Medak, Medak, Telengana	9 years	C/o college

P.R.GOVERNMENT COLLEGE (AUTONOMOUS), KAKINADA – 533 001

DEPARTMENT OF COMPUTER APPLICATIONS

ALLOCATION OF CREDITS AT SUBJECT LEVEL

Under CBCS from 2026-2027(Admitted Batch)

Course: LSC

For ALL B.Sc, B.Com and BA streams

YEAR	SEM	COURSE CODE	MODULE NATURE	COURSE TITLE	Hrs./ Week	Max. Marks			Credits
						Int.	Ext	Tot	
I	I	LSC1	LSC-1	AI Fundamentals	4	50	50	100	4
			LSC-1 Practical	AI Fundamentals-Practice Session	2	-	-	0	0
	II	LSC2	LSC-2	Applications of Artificial Intelligence	3	50	50	100	4
			LSC-2 Practical	Applications of Artificial Intelligence Practical	2	-	-	0	0

PITHAPUR RAJAH'S GOVERNMENT COLLEGE (AUTONOMOUS), KAKINADA – 533 001

DEPARTMENT OF COMPUTER APPLICATIONS

ALLOCATION OF CREDITS AT SUBJECT LEVEL

Under CBCS from 2026-2027(Admitted Batch) and Prior Batches

Course: B.Sc (Information Technology)

YEAR	SEM	COURSE CODE	MODULE NATURE	COURSE TITLE	Hrs./ Week	Max. Marks			Credits
						Int.	Ext	Tot	
I	I	26IT11	Course-1	Computer Fundamentals and Office Automation	3	50	50	100	3
			Course-1 Practical	Computer Fundamentals and Office Automation-Practical	2	-	-	50	1
		26IT12	Course-2	Problem Solving Using C	3	50	50	100	3
			Course-2 Practical	Problem Solving Using C - Practical	2	-	-	50	1
	II	26IT21	Course-3	Data Structures using C	3	50	50	100	3
			Course-3 Practical	Data Structures using C - Practical	2	-	-	50	1
		26IT22	Course-4	Digital Logic Design	3	50	50	100	3
			Course-4 Practical	Digital Logic Design-Practical	2	-	-	50	1
				Community Service Project	-	-	-	100	2
II	III	25IT31	Course-5	OOPS Through JAVA	3	50	50	100	3
			Course-5 Practical	OOPS Through JAVA-Practical	2	-	-	50	1
		25IT32	Course-6	Database Management Systems	3	50	50	100	3
			Course-6 Practical	Database Management Systems- Practical	2	-	-	50	1
		25IT33	Course-7	Computer Organisation	3	50	50	100	3
			Course-7 Practical	Computer Organisation Practical	2	-	-	50	1

	IV	25IT41	Course-8	Operating Systems	3	50	50	100	3
			Course-8 Practical	Operating Systems- Practical	2	-	-	50	1
		25IT42	Course-9	Computer Networks	3	50	50	100	3
			Course-9 Practical	Computer Networks- Practical	2	-	-	50	1
		25IT43	Course-10	Python Programming	3	50	50	100	3
			Course-10 Practical	Python Programming- Practical	2	-	-	50	1
				Short –term Internship	-	-	-	100	4
III	V	24IT51	Course-12	R Programming - (T)	3	50	50	100	3
			Course-12 Practical	R Programming - (P)	2	-	-	50	1
		24IT52	Course-13	Software Engineering- (T)	3	50	50	100	3
			Course-13 Practical	Software Engineering - (P)	2	-	-	50	1
		24IT53	Course-14	Computer Networks- (T)	3	50	50	100	3
			Course-14 Practical	Computer Networks - (P)	2	-	-	50	1
		24IT54	Course-15	Mobile Application Development –(T)	3	50	50	100	3
			Course-15 Practical	Mobile Application Development –(P)	2	-	-	50	1
	VI			Semester Apprenticeship	-	-	-	200	4

SEMESTER-1

P.R.GOVERNMENT COLLEGE(AUTONOMOUS), KAKINADA						
Course Code : 26IT11	COMPUTER FUNDAMENTALS & OFFICE AUTOMATION			I B.Sc (IT) SEM-I Major 1 2026-27		
Hours	90 (60+ 30)			L	T	P C
Pre requisites				3	-	1 4

Course Objectives	
C01	1. Understand foundational computing concepts including number systems, evolution of computers, and architectural components.
C02	2. Explore basic computer organization and network fundamentals, recognizing device functions, system types, and internet components.
C03	3. Demonstrate proficiency in word processing and presentation tools, applying formatting techniques and design elements for professional outputs.
C04	4. Develop competency in spreadsheet operations, employing formulas, charts, and data-handling techniques.
C05	5. Apply advanced data modeling and productivity features to analyze and visualize data efficiently using modern tools.

Course Outcomes

On completion of the course, the students will be able to		
Outcome	Description	Cognitive Level
C01	Convert between binary, decimal, octal, and hexadecimal systems	Knowledge
C02	Learners will demonstrate basic blocks of a computer and fundamental networking knowledge.	Knowledge
C03	Create professional-level documents and design visually appealing presentations	Analysis and Evaluation
C04	Manipulate data within spreadsheets, apply formulas	Application
C05	Apply data modeling techniques to analyze, organize, and represent	Creativity

P.R.GOV.T.COLLEGE (AUTONOMOUS), KAKINADA
DEPARTMENT OF COMPUTER APPLICATIONS
I B.Sc (IT) Semester-I (W.E.F.2026-27)
COMPUTER FUNDAMENTALS & OFFICE AUTOMATION

Unit 1. Number Systems, Evolution, Block Diagram and Generations:

Number Systems: Binary, Decimal, Octal, Hexadecimal; conversions between number systems.

Evolution of Computers: History from early mechanical devices to modern-day systems.

Block Diagram of a Computer: Components like Input Unit, Output Unit, Memory, CPU (ALU + CU).

Generations of Computers: First to Fifth Generation – technologies, characteristics, examples.

Unit 2. Basic organization and N/W fundamentals:

Computer Organization: Functional components – Input/ Output devices, Storage types, Memory hierarchy.

Types of Computers: Micro, Mini, Mainframe, and Supercomputers.

Networking Fundamentals: Definition, need for networks, types (LAN, WAN, MAN), topology (Star, Ring, Bus).

Internet Basics: IP Address, Domain Name, Web Browser, Email, WWW.

Unit 3. Word Processing and presentations:

Word Processing Basics: Using MS Word/Google Docs – formatting, styles, tables, mail merge.

Presentation Tools: Using PowerPoint/Google Slides – slide design, animations, transitions.

Applications: Creating resumes, reports, brochures, and presentations. Keyboard Shortcuts

Unit 4. Spreadsheet Basics:

Spreadsheet Concepts: Understanding rows, columns, cells in tools like MS Excel/Google Sheets, cell referencing.

Functions and Formulae: SUM, AVERAGE, IF, COUNT.

Charts and Graphs: Creating visual representations

Data Handling: Sorting, filtering, conditional formatting.

Text Functions: LEFT, RIGHT, MID, LEN, TRIM, CONCAT, TEXTJOIN

Advanced Functions: Logical: IF, AND, OR, IFERROR, Lookup: VLOOKUP, HLOOKUP, XLOOKUP, INDEX, MATCH

Unit 5. Data Analysis and Visualization:

Conditional Formatting: Custom rules, Color scales, Icon sets, Data bars

Data Analysis Tools: Pivot Tables and Pivot Charts, Data Validation (Drop-downs, Input Messages, Error Alerts), What-If Analysis: Goal Seek, Scenario Manager, Data Tables

Charts and Dashboards: Creating Interactive Dashboards, Using slicers with Pivot Tables, Combo Charts and Sparklines

Productivity Tips: Using Named Ranges, Freeze Panes, Split View

Textbooks:

1. Fundamentals of Computers, Reema Thareja, Oxford University Press, Second Edition
2. Fundamentals of Computers, V. Rajaraman – PHI Learning
3. Introduction to Computers by Peter Norton – McGraw Hill
4. Microsoft Office 365 In Practice by Randy Nordell – McGraw Hill Education

References:

1. Excel 2021 Bible by Michael Alexander, Richard Kusleika – Wiley
2. Networking All-in-One For Dummies by Doug Lowe – Wiley
3. Microsoft Official Docs and Training: <https://learn.microsoft.com>
4. Google Workspace Learning Center: <https://support.google.com/a/users/>
Activities:

Outcome: At the End of the Course, The Students will be able to explain different number systems, the historical evolution of computers, and identify key components in a block diagram.

Activity: Create a digital poster or infographic comparing number systems (binary, decimal, octal, hexadecimal) and illustrating the timeline of computer generations with key innovations.

Evaluation Method: Rubric-based assessment of the poster presentation on a 10-point scale focusing on:

- Accuracy of number system conversions
- Correct identification of block diagram components
- Visual organization and creativity

Outcome: Learners will demonstrate basic blocks of a computer and fundamental networking knowledge.

Activity: Design a concept map showing the internal architecture of a computer and types of networks (LAN, WAN, MAN), including devices and topologies.

Evaluation Method: Checklist-based peer review and instructor validation:

- Completeness of the map
- Correctness of networking concepts
- Use of appropriate terminology
- Logical flow and structure of the map

Outcome: Learners will create professional-level documents and design visually appealing presentations using word processing software and presentation software.

Activity: Prepare a formal report (e.g., project proposal) in a word processor and present it using a slide deck with transitions, embedded media, and design elements.

Evaluation Method: Performance-based evaluation using a 10-point scoring scale:

- Formatting and structure of the document
- Presentation aesthetics and clarity
- Communication skills during presentation

Outcome: Learners will manipulate data within spreadsheets, apply formulas, and generate accurate summaries and visualizations.

Activity: Analyze a dataset (e.g., student scores or sales data) using spreadsheet software. Apply formulas (SUM, AVERAGE, IF, VLOOKUP) and create relevant charts.

Evaluation Method: Practical test with a rubric:

- Correct use of formulas
- Accuracy of data summaries

Outcome: Learners will apply data modelling techniques to analyze, organize, and represent data effectively in various scenarios.

Activity: Prepare an interactive dashboard for a given data set using EXCEL.

Evaluation Method: Evaluation of the dashboard on a 10-point scoring scale:

- Presentation aesthetics and clarity
- Interactiveness
- Communication skills during presentation

P.R.GOV.T.COLLEGE (AUTONOMOUS), KAKINADA
DEPARTMENT OF COMPUTER APPLICATIONS
I B.Sc (IT) Semester-I (W.E.F.2026-27)
COMPUTER FUNDAMENTALS & OFFICE AUTOMATION

Practical

Credits: 1

2 hrs/week

List of Experiments:

1. Demonstration of Assembling and Desassembling of Computer Systems.
2. Identify and prepare notes on the type of Network topology of your institution.
3. Prepare your resume in Word.
4. Using Word, write a letter to your higher official seeking 10-days leave.
5. Prepare a presentation that contains text, audio and video.
6. Using a spreadsheet, prepare your class Time Table.
7. Using a Spreadsheet, calculate the Gross and Net salary of employees (Min 5) considering all the allowances.
8. Generate the class-wise and subject-wise results for a class of 20 students. Also generate the highest and lowest marks in each subject.
9. Using IF, AND, OR, and IFERROR to Automate Grade Evaluation.
 - a. Create a table of student scores in different subjects.
 - b. Use IF to assign grades (A/B/C/Fail).
 - c. Use IFERROR to handle missing scores or invalid data.
10. Employee Database Search Using VLOOKUP, HLOOKUP, XLOOKUP, INDEX, and MATCH
 - a. Create a database of employees (Name, ID, Department, Salary).
 - b. Implement VLOOKUP to search by employee ID.
 - c. Use HLOOKUP to extract department heads by role.
 - d. Apply XLOOKUP for more flexible searches.
 - e. Use INDEX + MATCH as an alternative to VLOOKUP.
11. Sales Report Analysis Using Pivot Tables and Charts
 - a. Use a dataset of product sales (Product, Region, Date, Quantity, Revenue).
 - b. Create Pivot Tables to summarize data by region/product.
 - c. Insert Pivot Charts for visual analysis (e.g., bar, line).
 - d. Add slicers to make the dashboard interactive.
12. Designing a Data Entry Form with Drop-downs and Input Rules
 - a. Create a student registration form.
 - b. Add drop-down lists for course selection using Data Validation.
 - c. Add input messages to guide users.

d. Add error alerts for wrong entries.

13. Monthly Budget Planning using Goal Seek and Scenario Manager

a. Create a simple personal budget (income, expenses, savings).

b. Use Goal Seek to determine income needed to save a desired amount.

c. Use Scenario Manager to compare different budgeting scenarios (best/ worst/ realistic case).

d. Create a one-variable Data Table to analyze how different expenses affect savings.

14. Dashboard Creation Using Combo Charts, Sparklines & Slicers

a. Use existing sales or attendance data.

b. Insert combo charts (e.g., column + line).

c. Add sparklines to show trends.

d. Use slicers with Pivot Tables to control dashboard elements.

e. Finalize and format for interactivity.

P.R.GOV.T.COLLEGE (AUTONOMOUS), KAKINADA

DEPARTMENT OF COMPUTER APPLICATIONS

I B.Sc (IT) Semester-I (W.E.F.2026-27)

COMPUTER FUNDAMENTALS & OFFICE AUTOMATION

Model blue print for the model paper and choice

S.NO	Type of Question	To be given in the Question Paper			To be answered		
		No. of Questions	Marks allotted to each question	Total Marks	No. of Questions	Marks allotted to each question	Total Marks
1	Section-I Essay Questions	6	10	60	3	10	30
2	Section-II Short Questions	7	5	35	4	5	20
TOTAL		13		95	TOTAL MARKS		50

Model Blue print for the question paper Setter

Chapter Name	Essay Questions 10 Marks	Short Questions 5 Marks	Marks allotted to the chapter
UNIT-I	1	2	20
UNIT-II	1	2	20
UNIT-III	2	1	25
UNIT-IV	1	1	15
UNIT-V	1	1	15
Total No. of questions	6	7	
Total Marks Including choice			95

P.R.GOV.T.COLLEGE (AUTONOMOUS), KAKINADA

DEPARTMENT OF COMPUTER APPLICATIONS

I B.Sc (IT) Semester-I (W.E.F.2026-27)

COMPUTER FUNDAMENTALS & OFFICE AUTOMATION

Time: 2 Hrs

Max. Marks: 50

SECTION - I

Answer any three of the following questions. Must attempt at least one question from each part.

Each question carries 10 Marks.

3 X 10 = 30M

Part-A

1. Explain the generations of Computers and their technology.
2. What are the different types of computers? Explain each type.
3. What is MS-Word? Explain features of MS –Word?.

Part-B

4. How to create presentation in MS-Power point?
5. What are the different types of charts that are available in MS-Excel? Explain.
6. Describe the purpose of data analysis tools.

SECTION - II

Answer any four of the following questions. Each question carries 5 marks. 4 X 5 = 20M

7. How to convert decimal number to binary number ? explain with example
8. Explain about Memory in Computer?
9. Explain about network Topologies?
10. Describe working of E-mail.
11. How to insert table in Microsoft word?
12. What are the types of Cell Referencing? Explain
13. Write about Pivot tables.

P.R.GOV.T.COLLEGE (AUTONOMOUS), KAKINADA
DEPARTMENT OF COMPUTER APPLICATIONS
I B.Sc (IT) Semester-I (W.E.F.2026-27)
COMPUTER FUNDAMENTALS & OFFICE AUTOMATION

Question Bank

Unit-1

Essay Questions

1.	Draw the Block Diagram of Digital Computer? Explain the functions of each component?	BT2	C01	P01
2.	Discuss the Five generations of computer and specify the underlying technology?	BT1	C02	P02
3.	Explain the Four number systems with examples?	BT1	C03	P03

Unit-2

1.	Classify the types of computers based on their size and capability?	BT1	C01	P01
2.	Define the Network and Explain Network Topologies	BT1	C02	P02
3.	Define the Memory hierarchy? Differentiate Primary and Secondary memory?	BT2	C03	P03

Unit-3

1.	What is MS-Word? Explain features of MS –Word?.	BT1	C01	P01
2.	What is Mail Merge? Explain the Process of Creating a mail merge	BT1	C02	P02
3.	What is PowerPoint? How to create presentation in MS-Power point?	BT1	C02	P02

Unit-4

1.	Discuss the importance of Charts and Graphs in Excel?	BT1	C01	P01
2.	Explain about Sorting and Filtering in Excel?	BT1	C02	P02
3.	Briefly Explain about SUM, AVERAGE, IF, MAX, MIN functions in Excel	BT2	C03	P03

Unit-5

1.	Explain What-IF-Analysis?	BT1	C01	P01
2.	Explain about Pivot Table in excel?	BT1	C02	P02

Short Answer Questions

Unit-1

1.	Briefly differentiate between roles of ALU and CU	BT2	C01	P01
2.	Define Hexadecimal number System and list out the digits / characters	BT1	C02	P02
3.	Explain about Memory in Computer?	BT1	C03	P03
4.		BT3	C01	P01

Unit-2

1.	Differentiate LAN vs WAN	BT2	C01	P01
2.	What is Email? Write its advantages	BT2	C02	P02
3.	Define IP Address?	BT1	C03	P03

Unit-3

1.	What is Header and Footer?	BT1	C01	P01
2.	Differentiate Transition vs Animation?	BT2	C02	P02
3.	Briefly Explain about any Five Shortcut keys in MS-Word	BT2	C02	P02

Unit-4

1.	What is Cell referencing in excel?	BT2	C01	P01
2.	What is Conditional Formatting?	BT1	C02	P02

Unit-5

1.	Explain about Data Validation?	BT2	C01	P01
2.	Explain the purpose of Freeze pane and Split view?	BT1	C02	P02

P.R.GOVERNMENT COLLEGE(AUTONOMOUS), KAKINADA					
Course Code : 26IT12	PROBLEM SOLVING USING C	I B.Sc (IT) SEM-I Major 2 2026-27			
Hours	90 (60+ 30)	L	T	P	C
Pre requisites		3	-	1	4

Course Objective
1. Understand the basics of C 2. Understand the syntax of C-programming language 3. Apply C-programming skill to solve problems

Course Outcomes	
Upon successful completion of the course, a student will be able to:	
C01	1. Understand the functionality of a Digital Computer and fundamental constructs of programming.
C02	2. Analyze and develop solutions to a given problem using control statements.
C03	3. Work with arrays and textual information
C04	4. Understand the concept of functional hierarchical code organization.
C05	5. Gain knowledge on derived data types and file handling.

P.R.GOV.T.COLLEGE(AUTONOMOUS),KAKINADA
DEPARTMENT OF COMPUTER APPLICATIONS
I B.Sc (IT) Semester-I (W.E.F.2026-27)
PROBLEM SOLVING USING C

Unit 1. Introduction to computer programming:

Introduction, Types of software, Compiler and interpreter, Concepts of Machine level, Assembly level and high-level programming, Flowcharts and Algorithms, Fundamentals of C: History of C, Features of C, C Tokens-variables and keywords and identifiers, constants and Data types, Rules for constructing variable names, Operators, Structure of C program, Input /output statements in C-Formatted and Unformatted I/O

Unit 2. Control statements:

Decision making statements: if, if else, else if ladder, switch statements. Loop control statements: while loop, for loop and do-while loop. Jump Control statements: break, continue and goto.

Unit 3. Derived data types in C:

Arrays: One Dimensional arrays - Declaration, Initialization and Memory representation; Two Dimensional arrays -Declaration, Initialization and Memory representation. Strings: Declaring & Initializing string variables; String handling functions, Character handling functions

Unit 4. Functions:

Pointers: Pointer data type, Pointer declaration, initialization, accessing values using pointers. Pointer arithmetic, Pointers and arrays. Function Prototype, definition and calling. Return statement. Nesting of functions. Categories of functions. Recursion (Basic Concept only). Parameter Passing by address & by value. Local and Global variables. Storage classes: automatic, external, static and register.

Unit 5. Dynamic Memory Management:

Introduction, Functions-malloc, calloc, realloc, free Structures: Basics of structure, structure members, accessing structure members, nested structures, array of structures, structure and functions, structures and pointers. Unions - Union definition; difference between Structures and Unions. Working with text files - modes: opening, reading, writing and closing text files.

Text Books:

1. Programming in ANSI C, E. Balagurusamy, Tata McGraw Hill, 6 thEdn,
2. Computer fundamentals and programming in C, Reema Theraja, Oxford University Press

Reference Books:

1. Let us C, Y Kanetkar, BPB publications
2. Head First C: A Brain-Friendly Guide, David Griffiths, Dawn Griffiths

Activities:

Outcome: Understand basic computing concepts, programming paradigms and write structured C programs.

Activity: Create a concept map of computing fundamentals and programming paradigms (procedural, structured, object-oriented). Then, they write a structured C program (e.g., a calculator or student grade system) using proper syntax, indentation, and modular design.

Evaluation Method: Rubric-based Code Review & Viva to check the

- The correctness of the concept map
- Correct use of structure (main + functions)
- Identification of paradigm used
- Code readability and documentation

Outcome: Apply control flow statements to solve logical and repetitive tasks in C.

Activity: Implement a program that solves a logic puzzle (e.g., number guessing game, pattern generation, or prime number finder) using if, switch, for, while, and do-while.

Evaluation Method: Automated Test Cases + Peer Review to check the

- Correct use of control statements
- Logical correctness of output
- Efficiency and edge case handling
- Peer feedback on clarity and logic

Outcome: Implement arrays and string operations to manage and manipulate data efficiently.

Activity: Build a program that stores and arranges student marks in ascending and descending order using arrays and performs string operations like concatenation, comparing, and formatting names.

Evaluation Method: Functional Demonstration + Code Walkthrough to check the

- Correct array and string usage
- Memory efficiency
- Handling of invalid inputs
- Explanation of sorting/searching logic

Activity:

- Recursive Problem Solver Students write a modular program to solve a recursive problem (e.g., factorial, Fibonacci, or Tower of Hanoi) using functions with parameters and return values.

Evaluation Method:

- Code Trace + Written Quiz
- Correct function decomposition
- Proper parameter passing (by value/reference)
- Recursion depth and base case handling
- Quiz on tracing recursive calls

Outcome: Utilize pointers and memory operations for effective data handling. Demonstrate competence in dynamic memory allocation and text file processing.

Activity: Create a program that dynamically stores user input (e.g., survey responses) using pointers and writes/reads the data to/from a text file.

Evaluation Method: Memory Debugging + File I/O Assessment to check the

- Proper use of malloc, calloc, realloc, and free
- Pointer arithmetic and dereferencing
- File creation, reading, writing, and error handling
- Use of tools like Valgrind or manual memory trace (Optional for Unix flavours)

P.R.GOV.T.COLLEGE (AUTONOMOUS), KAKINADA
DEPARTMENT OF COMPUTER APPLICATIONS
I B.Sc (IT) Semester-I (W.E.F.2026-27)
PROBLEM SOLVING USING C

Practical

Credits: 1 2 hrs/week

List of Experiments:

1. Write a program to check whether the given number is Armstrong or not.
2. Write a program to find the sum of individual digits of a positive integer.
3. Write a program to generate the first n terms of the Fibonacci sequence.
4. Write a program to find both the largest and smallest number in a list of integer values
5. Write a program to demonstrate change in parameter values while swapping two integer variables using Call by Value & Call by Address
6. Write a program to perform various string operations.
7. Write a program to search an element in a given list of values.
8. Write a program that uses functions to add two matrices.
9. Write a program to calculate factorial of given integer value using recursive functions
10. Write a program for multiplication of two N X N matrices.
11. Write a program to sort a given list of integers in ascending order.
12. Write a program to calculate the salaries of all employees using Employee (ID, Name, Designation, Basic Pay, DA, HRA, Gross Salary, Deduction, Net Salary) structure.
 - a. DA is 30 % of Basic Pay
 - b. HRA is 15% of Basic Pay
 - c. Deduction is 10% of (Basic Pay + DA)
 - d. Gross Salary = Basic Pay + DA+ HRA
 - e. Net Salary = Gross Salary - Deduction
13. Write a program to read / write the data from / to a file.
14. Write a program to reverse the contents of a file and store in another file.
15. Write a program to create Book (ISBN, Title, Author, Price, Pages, Publisher) structure and store book details in a file and perform the following operations
 - a. Add book details
 - b. Search a book details for a given ISBN and display book details, if available
 - c. Update a book details using ISBN
 - d. Delete book details for a given ISBN and display list of remaining Books.

P.R.GOV.T.COLLEGE (AUTONOMOUS), KAKINADA
DEPARTMENT OF COMPUTER APPLICATIONS
I B.Sc (IT) Semester-I (W.E.F.2026-27)

PROBLEM SOLVING USING C

Model blue print for the model paper and choice

S.NO	Type of Question	To be given in the Question Paper			To be answered		
		No. of Questions	Marks allotted to each question	Total Marks	No. of Questions	Marks allotted to each question	Total Marks
1	Section-I Essay Questions	6	10	60	3	10	30
2	Section-II Short Questions	7	5	35	4	5	20
TOTAL		13		95	TOTAL MARKS		50

Model Blue print for the question paper Setter

Chapter Name	Essay Questions 10 Marks	Short Questions 5 Marks	Marks allotted to the chapter
UNIT-I	1	2	20
UNIT-II	1	2	20
UNIT-III	2	1	25
UNIT-IV	1	1	15
UNIT-V	1	1	15
Total No. of questions	6	7	
Total Marks Including choice			95

P.R.GOV.T.COLLEGE (AUTONOMOUS), KAKINADA

DEPARTMENT OF COMPUTER APPLICATIONS

I B.Sc (IT) Semester-I (W.E.F.2026-27)

PROBLEM SOLVING USING C

MODEL PAPER

Time : 2Hrs

SEMESTER-I

Max. Marks: 50

SECTION-I

Answer Any Three Questions. At least One question from each Part

(3x10=30M)

Part-A

1. Define Algorithm? Explain the features of an Algorithm?BT-2
2. Explain various operators in C with example. BT-1
3. Explain various Looping Statements in C with example. BT-1

Part-B

4. Write about Two Dimensional Array? BT-1
5. What is Function ? Explain about Function call?BT-2
6. Write difference between structures and unions?BT-3

SECTION-II

Answer any FOUR Questions. Each question carries 5 marks

(4x5=20M)

7. Write about features of C language.BT-2
8. Explain various data types in C.BT-2
9. Write about one dimensional array with example.BT-1
10. Write about break and continue statements with examples.BT-2
11. What is recursion? What advantage is there in its use?BT-1
12. Write about switch statements.BT-1
13. Explain about Dynamic Memory Allocation.BT-2

Question Bank

Essay Questions:

UNIT-I

1.	Define data type? Explain different data types in C-language	BT-1	CO1	P01
2.	List various Operators in C with example.	BT-2	CO1	P01
3.	What is an algorithm? What are the features of an Algorithm	BT-3	CO1	P01

UNIT-II

1.	Explain various Looping Statements in C with example.	BT-1	CO1	P01
2.	What is Jump Statement? Explain different Jumping Statements?	BT-2	CO2	P02
3.	Classify various Conditional Control Statements in C with example.	BT-4	CO1	P01

UNIT-III

1.	Explain various String handling Functions in C.	BT-3	CO3	P03
2.	Define Array? Explain 2-D array with example?	BT-1	CO3	P03
3.	Define Array? Explain 1-D array with example?	BT-2	CO3	P03

UNIT-IV

1.	What is a function? Explain in detail.	BT-1	CO4	P04
2.	Define Pointer? How can you declare, initialize, and accessing values using pointer?	BT-2	CO4	P04
3.	Define Recursion? How Recursion will be implemented?	BT-3	CO3	P04

UNIT-V

1.	Define Structure? Write about Basics of Structures?	BT-2	CO5	P05
2.	Write about Union? Explain with an Example?	BT-1	CO5	P05
3.	What is File? fopen() function with example	BT-4	CO5	P05

Short Questions:

UNIT-I

1.	Differentiate System Software and Application Software	BT-1	CO1	P01
2.	What is Algorithm? Given an Example	BT-1	CO1	P02
3.	Define Flow Chart? Given an Example	BT-4	CO1	P01

UNIT-II

1.	Explain Switch case in C-programming	BT-1	C01	P02
2.	Difference between Break & Continue Statements in C	BT-1	C02	P03
3.	Explain For – Loop statement in C	BT-1	C02	P02

UNIT-III

1.	List the Advantages and dis-Advantages of an Array	BT-1	C03	P02
2.	Difference between Fundamental and Derived data types in C	BT-4	C03	P02

UNIT-IV

1.	Define Function and its types in C	BT-1	C03	P03
2.	Explain Functions with No arguments and No return type?	BT-3	C04	P03

UNIT-V

1.	Write about Nested Structure in Detail?	BT-1	CO4	PO4
2.	Differentiate Structure and Union?	BT-2	CO4	PO3
3.	Explain Reading the Data from Files?	BT-1	CO4	PO4

Mapping of Cos with POs/PSOs

[illegible]

SEMESTER-2

P.R.GOVERNMENT COLLEGE(AUTONOMOUS), KAKINADA						
Course Code : 26IT21	DATA STRUCTURES USING C			I B.Sc (IT) SEM-II Major 3 2026-27		
Hours	90 (60+ 30)			L	T	P C
Pre requisites	Understanding of C Programming			3	-	1 4

Course Objective
1. Understand fundamental concepts of algorithms and data structures with focus on complexity analysis and abstract data types.
2. Explore various types of linked lists and their dynamic memory representations and operations.
3. Analyze and implement linear data structures, such as stacks and queues, and examine their real-world applications.
4. Apply sorting and searching algorithms, understanding their performance implications and optimization strategies.
5. Design and manipulate hierarchical and graph-based structures, applying traversal algorithms and understanding their practical uses in computing.

Course Outcomes	
Upon successful completion of the course, a student will be able to:	
C01	1. Explain algorithm characteristics, time and space complexity, and asymptotic notations with clarity.
C02	2. Implement and analyze different types of linked lists, including insertion, deletion, and traversal operations.
C03	3. Develop stack and queue data structures using arrays and linked lists, and apply them in expression evaluation.
C04	4. Apply efficient searching and sorting algorithms to solve computational problems and evaluate performance trade-offs.
C05	5. Construct and traverse tree and graph structures, using them to solve problems like shortest path and spanning trees.

P.R.GOV.T.COLLEGE (AUTONOMOUS), KAKINADA

DEPARTMENT OF COMPUTER APPLICATIONS

I B.Sc (IT) Semester-II (W.E.F.2026-27)

DATA STRUCTURES USING C

Unit 1. Basic Concepts: Algorithm: Definition and characteristics, Complexity analysis: Space Complexity, Time Complexity, Asymptotic Notations. Introduction to Data structures: Definition, Types of Data structures, Abstract Data Types (ADT), Introduction to Linked Lists, Representation of linked lists in Memory, Comparison between Linked List and Array.

Unit 2. Linked Lists: Types of Linked Lists - Singly Linked list, Doubly Linked list, Circularly Singly Linked list, Circularly Doubly Linked list; Implementation of Single Linked List ADT: Creating a List, Traversing a linked list, Searching in linked list, Insertion and deletion into linked list (At first Node, Specified Position, Last node).

Unit 3. Stacks and Queues: Introduction to stack ADT, Implementation of stacks using array and Linked List, Application of stacks - Polish Notations - Converting Infix to Post Fix Notation - Evaluation of Post Fix Notation. Queues: Introduction to Queue ADT, Implementation of Queues using array and Linked List, Application of Queues Types of Queues- Circular Queues, De-queues, Priority Queue, Heaps.

Unit 4. Searching and Sorting: Linear or Sequential Search, Binary Search, Hashing and collision resolution. Sorting: Selection Sort, Bubble Sort, Insertion Sort, Quick Sort and Merge Sort

Unit 5. Trees and Graphs: Tree Terminology, Binary Tree Representation, Traversal techniques, Expression Tree, Binary Search Tree- Definition, Operations on a Binary Search Tree: Creation, Search, Insertion & deletion. Graphs: Introduction to Graphs, Terminology, Representation (Adjacency Matrix, Adjacency List), Traversal of Graphs (DFS, BFS), Applications of Graphs, Concept of Shortest Path Problems, Concept of Minimum Cost Spanning Tree

Textbooks: 1. Data Structures Using C, Balagurusamy E. Tata McGraw Hill
2. Data Structures using C, Reema Thareja, Third Edition, Oxford University Press

Reference Books: 1. Data Structures, Lipschutz, Schaum's Outline Series, Tata Mcgraw-hill
2. Data Structures Using C, Ch. Vijay Kumar, Pen Press International

Activities: Outcome: Explain algorithm characteristics, time and space complexity, and asymptotic notations with clarity

Activity: Create a comparative chart of algorithms with different notations related to time and space complexities. Evaluation Method: Rubric-based assessment of the chart for correctness, clarity, and depth of explanation on a 10-point scale. Outcome: Implement and analyze different types of linked lists, including insertion, deletion, and traversal operations

Activity: Code a menu-driven program in C to implement single linked lists with all basic operations. Evaluation Method: Practical lab assessment with test cases and Viva-style questioning to explain pointer manipulation. Outcome: Develop stack and queue data structures using arrays and linked lists, and apply them in expression evaluation

Activity: Build a program to convert infix expressions to postfix and evaluate them using stacks;

Implement queues using both arrays and linked lists with enqueue/dequeue operations. Evaluation Method: Code review and execution of programs for sample cases and evaluation based on correctness and efficiency. Outcome: Apply efficient searching and sorting algorithms to solve computational problems and evaluate performance trade-offs

Activity: Implement and compare sorting algorithms (e.g., selection sort and bubble sort) and searching algorithms (e.g., Linear vs. Binary Search) on datasets of varying sizes. Record number of swaps and iterations for preparing a chart to assimilate the results. Evaluation Method: Performance report with graphs and analysis. Oral presentation or peer review discussing trade-offs and algorithm selection rationale.

Outcome: Construct and traverse tree and graph structures, using them to solve problems like shortest path and spanning trees

Activity: Implement binary trees and graphs using adjacency lists/matrices. Evaluation Method: Lab demo with sample inputs and visual output (e.g., tree traversal order, graph paths).

P.R.GOV.T.COLLEGE (AUTONOMOUS), KAKINADA

DEPARTMENT OF COMPUTER APPLICATIONS

I B.Sc (IT) Semester-II (W.E.F.2026-27)

DATA STRUCTURES USING C

Practical

Credits: 1

2 hrs/week

List of Experiments

1. Write a program to read 'N' numbers of elements into an array and also perform the following operation on an array
 - a. Add an element at the beginning of an array
 - b. Insert an element at given index of array
 - c. Update an element using a values and index
 - d. Delete an existing element
2. Write a program to implement Single Linked List with insertion, deletion and traversal operations3. Write a program to implement Doubly Linked List with insertion, deletion and traversal operations
4. Write a program to implement the Stack operations using Arrays and Linked Lists.
5. Write a program to convert a given infix expression to a postfix expression using stacks.
6. Write a program to implement the Queue operations using Arrays and Linked Lists.
7. Write a program to implement the Circular Queue operations using Arrays.
8. Write a program for Binary Search Tree Traversals
9. Write a program to search an item in a given list using the following Searching Algorithms
 - a. Linear Search
 - b. Binary Search.
10. Write a program for implementation of the following Sorting Algorithms
 - a. Bubble Sort
 - b. Insertion Sort
 - c. Quick Sort
 - d. Merge Sort

P.R.GOV.T.COLLEGE (AUTONOMOUS), KAKINADA
DEPARTMENT OF COMPUTER APPLICATIONS
I B.Sc (IT) Semester-II (W.E.F.2026-27)

DATA STRUCTURES USING C
Model blue print for the model paper and choice

S.NO	Type of Question	To be given in the Question Paper			To be answered		
		No. of Questions	Marks allotted to each question	Total Marks	No. of Questions	Marks allotted to each question	Total Marks
1	Section-I Essay Questions	6	10	60	3	10	30
2	Section-II Short Questions	7	5	35	4	5	20
TOTAL		13		95	TOTAL MARKS		50

Model Blue print for the question paper Setter

Chapter Name	Essay Questions 10 Marks	Short Questions 5 Marks	Marks allotted to the chapter
UNIT-I	1	2	20
UNIT-II	1	2	20
UNIT-III	2	1	25
UNIT-IV	1	1	15
UNIT-V	1	1	15
Total No. of questions	6	7	
Total Marks Including choice			95

P.R.GOV.T.COLLEGE (AUTONOMOUS), KAKINADA
DEPARTMENT OF COMPUTER APPLICATIONS
I B.Sc (IT) Semester-II (W.E.F.2026-27)
DATA STRUCTURES USING C

Time: 2 Hrs

Max. Marks: 50

SECTION - I

Answer any three of the following questions. Must attempt at least one question from each part.

Each question carries 10 Marks.

3 X 10 = 30M

Part-A

1. Define Data Structures. Describe different types of data structures?
2. What is an Abstract Data type (ADT)? Explain its importance with examples
3. What is Linked list? Explain different types of linked list?

Part-B

4. What is Stack ? Explain implementation stack using array?
5. Explain working of binary search algorithm with an example
6. Describe the Operations on a binary search tree (creation, insertion, search)

SECTION - II

Answer any four of the following questions. Each question carries 5 marks. 4 X 5 = 20M

7. Differentiate between time complexity and space complexity
8. Compare linked list and array
9. What is single linked list
10. What is Queue? Give an example?
11. Differentiate between linear and binary search?
12. What is hashing?
13. Define tree terminology.

PRGOVERNMENTCOLLEGE(AUTONOMOUS),KAKINADA						
Course Code 26IT22	DIGITAL LOGIC DESIGN			I B.Sc (IT) SEM-II Major-4 2026-27		
Hours	90 (60Theory+ 30Practical)			L	T	P C
Pre requisites	BASIC COMPUTER KNOWLEDGE			3	-	1 4

Course Objective
1. Introduce the fundamentals of number systems, their conversions, and binary arithmetic operations.
2. Simplify logic functions using Boolean algebra, Karnaugh maps, and universal gates.
3. Develop proficiency in designing basic combinational circuits like adders and subtractors.
4. Equip students with the skills to implement advanced combinational components such as multiplexers, encoders, and decoders.
5. Foster understanding of sequential circuits, flip-flops, counters, and shift registers for system-level design.

Course Outcomes		
On Completion of the course, the students will be able to-		Cognitive Domain
CO1	Apply concepts of number systems to perform radix conversions and binary arithmetic using signed and unsigned formats.	Understanding
CO2	Simplify logic functions using Boolean algebra, Karnaugh maps, and universal gates.	Application
CO3	Design and analyze combinational circuits such as half adders, full adders, and subtractors.	Application
CO4	Construct advanced combinational logic modules, including multiplexers, demultiplexers, encoders, decoders, and their hierarchical versions. Realize complex Boolean functions using combinations of logic modules.	Application
CO5	Develop and evaluate sequential circuits such as flip-flops, latches, counters, and shift registers.	Application

P R GOVT COLLEGE (AUTONOMOUS), KAKINADA
DEPARTMENT OF COMPUTER APPLICATIONS
I BSC (IT) SEMESTER - II (W.E.F. 2026-27)
SEMESTER-II
COURSE 4: DIGITAL LOGIC DESIGN

Unit 1: Number Systems:

Conversion of numbers from one radix to another radix, r 's, $(r-1)$'s complements, signed binary numbers, addition and subtraction of unsigned and signed numbers, weighted and unweighted codes.

Unit 2. Logic Gates and Boolean Algebra:

NOT, AND, OR, universal gates, X-OR and X-NOR gates, Boolean laws and theorems, complement and dual of a logic function, canonical and standard forms, two level realization of logic functions using universal gates, minimizations of logic functions (POS and SOP) using Boolean theorems, K-map (up to four variables), don't care conditions.

Unit 3. Combinational Logic Circuits – 1:

Design of half adder, full adder, half subtractor, full subtractor, ripple adders and subtractors, ripple adder / subtractor.

Unit 4. Combinational Logic Circuits – 2:

Design of decoders, encoders, priority encoder, multiplexers, demultiplexers, higher order decoders, demultiplexers and multiplexers, realization of Boolean functions using decoders, multiplexers.

Unit 5. Sequential Logic Circuits:

Classification of sequential circuits, latch and flip-flop, RS- latch using NAND and NOR Gates RS, JK, T and D flip-flops, truth tables and excitation tables, conversion of flip- flops, flip-flops with asynchronous inputs (preset and clear). Registers- shift registers, bidirectional shift registers, universal shift register, design of ripple counters, modulus counters.

Text Books:

1. Digital Design, M. Morris Mano, Michael D Ciletti, 5th edition, Pearson.
2. Digital Logic Design, K.C. Rao, Ramana, Pen International Press

Reference Books:

1. Digital Electronics and Logic Design, Jaydeep Chakravorty, Universities Press
2. Digital Logic Design, Sonali Singh, BPB Publications

Activities:

Outcome: Apply concepts of number systems to perform radix conversions and binary arithmetic using signed and unsigned formats

Activity: Design a calculator in a spreadsheet or simulation tool (e.g., Logisim) that performs: Decimal $\leftrightarrow$ Binary $\leftrightarrow$ Hexadecimal conversions and binary arithmetic (addition, subtraction).

Evaluation Method: Rubric-based evaluation on a 10-point scale (conversion accuracy, arithmetic correctness)

Outcome: Simplify logic functions using Boolean algebra, Karnaugh maps, and universal gates

Activity: Provide students with complex Boolean expressions and truth tables. Ask them to: Simplify using Boolean laws, Minimize using Karnaugh maps and Implement using only NAND or NOR gates

Evaluation Method: Worksheet submission with step-by-step simplification and evaluation of gate-level implementation using a 10-point scale.

Outcome: Design and analyze combinational circuits such as half adders, full adders, and Subtractors

Activity: Build and simulate: Half adder and full adder using logic gates, and half and full subtractor circuits

Evaluation Method: Evaluate the correctness of the circuits for different inputs on a 10-point scale.

Outcome: Construct advanced combinational circuits, including multiplexers, demultiplexers, encoders and decoders.

Activity: Design Multiplexers for function selection, Decoders for control signal generation and Encoders for input compression

Evaluation Method: Project-based evaluation with functional demo and assessments based on a 10-point scale.

Outcome: Develop and evaluate sequential circuits such as flip-flops, latches, counters, and shift registers

Activity: Implement and test SR, JK, D, T flip-flops, asynchronous and synchronous counters using a simulator (E.g. Logisim, Multisim)

Evaluation Method: Lab assessment on a 10-point scale to understand the correctness of the circuit and presentation of the design.

P R GOVT COLLEGE (AUTONOMOUS), KAKINADA

DEPARTMENT OF COMPUTER APPLICATIONS

I BSC (IT) SEMESTER - II (W.E.F. 2026-27)

SEMESTER-II

COURSE 4: DIGITAL LOGIC DESIGN

Practical

Credits: 1

2 hrs/week

List of Experiments

The laboratory work can be done by using physical gates and necessary equipment or simulators.

Simulators: <https://sourceforge.net/projects/gatesim/> or <https://circuitverse.org/> or any free opensource simulator

1. Introduction to digital electronics lab- nomenclature of digital ICs, specifications, study of the data sheet, concept of Vcc and ground, verification of the truth tables of logic gates using TTL ICs.
2. Implementation of the given Boolean functions using logic gates in both SOP and POS forms
3. Realization of basic gates using universal gates.
4. Design and implementation of half and full adder circuits using logic gates.
5. Design and implementation of half and full subtractor circuits using logic gates.
6. Verification of stable tables of RS, JK, T and D flip-flops using NAND gates.
7. Implementation and verification of Decoder and encoder using logic gates.
8. Implementation of 4X1 MUX and DeMUX using logic gates.
9. Implementation of 8X1 MUX using suitable lower order MUX.
10. Implementation of 7-segment decoder circuit.
11. Implementation of 4-bit parallel adder.
12. Design and verification of 4-bit modulus counter

P R GOVT COLLEGE (AUTONOMOUS), KAKINADA
DEPARTMENT OF COMPUTER APPLICATIONS
I BSC (IT) SEMESTER - II (W.E.F. 2026-27)
DIGITAL LOGIC DESIGN

PAPER-4

Marks 50M

Model blue print for the model paper and choice

S.NO	Type of Question	To be given in the Question Paper			To be answered		
		No. of Questions	Marks allotted to each question	Total Marks	No. of Questions	Marks allotted to each question	Total Marks
1	Section-A Essay Questions	6	10	60	3	10	30
2	Section-B Short Questions	7	5	35	4	5	20
TOTAL		13		95	TOTAL MARKS		50

$$\text{Percentage of choice given} = \frac{95-50}{95} \times 100 = \frac{45}{95} \times 100 = 47.36\%$$

Model Blue print for the question paper setter

Chapter Name	Essay Questions 10 Marks	Short Questions 5 Marks	Marks allotted to the chapter
UNIT-I	2	2	30
UNIT-II	1	1	15
UNIT-III	1	1	15
UNIT-IV	1	1	20
UNIT-V	1	2	15
Total No. of questions	6	7	
Total Marks Including choice			95

P R GOVT COLLEGE (AUTONOMOUS), KAKINADA
DEPARTMENT OF COMPUTER APPLICATIONS
I BSC (IT) SEMESTER - II (W.E.F. 2026-27
DIGITAL LOGIC DESIGN

PAPER-

Marks 50M

Time:2 Hrs

SEMESTER-II

Max.Marks:50

SECTION-I

Answer Any Three Questions. At least One question from each part

(3x10=30M)

Part-A

1. Explain the various types of number systems with examples.
2. Discuss signed binary number representation and the methods of addition and subtraction for signed numbers.
3. Explain the basic logic gates with their truth tables and symbols.

Part-B

4. Explain the working of a half subtractor and full subtractor with truth tables and logic diagrams
5. Explain the working of a multiplexer and demultiplexer with truth tables and diagrams.
6. Explain the classification of sequential circuits with examples.

SECTION-II

Answer any FOUR Questions. Each question carries 5 marks

(4x5=20M)

7. Convert $(101101)_2$ to decimal, octal, and hexadecimal.?
8. Convert $(256)_{10}$ into binary and hexadecimal?
9. What are universal gates? Why are they called universal?
10. What is the function of a half adder?
11. Define encoder and decoder.
12. What is the difference between multiplexer and demultiplexer?
13. What is the difference between combinational and sequential circuits?

P.R. GOVT.COLLEGE (AUTONOMOUS), KAKINADA
DEPARTMENT OF COMPUTER APPLICATIONS
I B.Sc. – IT Semester – II (W.E.F. 2026-27)
DIGITAL LOGIC DESIGN

Question Bank

Unit-1

Essay Questions

1.	Describe classification of Codes? Explain weighted and unweighted codes with examples (BCD, Gray, Excess-3 code).	BT2	CO1	PO1
2.	Describe r 's complement and $(r-1)$'s complement of unsigned subtraction using 9's and 10's complement.	BT2	CO2	PO2
3.	Perform addition and subtraction of signed binary numbers using 2's complement method.	BT2	CO3	PO3

Unit-2

1.	Explain basic logic gates (NOT, AND, OR) and universal gates with truth tables	BT2	CO1	PO1
2.	Explain K-Map and simplify Boolean expressions $F(A,B,C,D) = \sum (0,2,5,8,10,13,15) + d(1,9,11)$ using K-map including don't care conditions.	BT1	CO2	PO2
3.	Prove Boolean laws and theorems using examples.	BT1	CO2	PO2

Unit-3

1.	Design and explain half adder and full adder with truth tables and logic diagrams.	BT1	CO1	PO1
2.	Design a ripple adder/subtractor circuit.	BT1	CO1	PO1

Unit-4

1	Design and explain decoder and encoder circuits with examples.	BT1	CO1	PO1
2	Explain multiplexer and demultiplexer with diagrams and applications.	BT1	CO2	PO2
3	Design a priority encoder and explain its operation.	BT1	CO2	PO2

Unit-5

1.	Describe RS, JK, D and T flip-flops with truth tables.	BT2	CO1	PO1
2.	Design ripple counters and modulus counters.	BT3	CO2	PO2

Unit-1

1.	Explain conversion of numbers from one radix to another. Convert $(101101)_2$ to decimal and hexadecimal.	BT2	CO1	PO1
2.	Explain 1's and 2's complement. Find 1's and 2's complement of a given binary number $(1010)_2$.	BT1	CO2	PO2
3.	Using 8-bit 1's complement subtract 14 from 25 and Add -25 to +14.	BT1	CO3	PO3

Unit-2

1.	Write truth tables for XOR and XNOR gates.	BT2	CO1	PO1
2.	Explain SOP and POS forms with examples.	BT1	CO3	PO3
3.	Explain dual of a logic function.	BT1	CO3	PO3
4.	Simplify Boolean expressions $F(A,B,C,D) = \sum (1,3,5,7,9,11,13,15)$ using K-map			

Unit-3

1.	Define ripple carry adder.	BT2	CO1	PO1
2.	4. Difference between half subtractor and full subtractor.	BT2	CO2	PO2

Unit-4

1.	Define multiplexer and its uses.	BT2	CO1	PO1
2.	Difference between encoder and decoder.	BT2	CO2	PO2
3.	What is demultiplexer?	BT2	CO3	PO3

Unit-5

1.	What is a shift register?	BT2	CO1	PO1
2.	Define RS latch using NAND gates.	BT2	CO2	PO2
3.	Explain RS flip-flop?	BT2	CO1	PO1
4.	Explain JK flip-flop?	BT2	CO2	PO2

Mapping of Cos with POs/PSOs

[illegible]

SEMESTER-3

P R GOVERNMENT COLLEGE (AUTONOMOUS), KAKINADA					
Course Code: 25IT31	OOPS Through JAVA	II B.Sc (IT) Sem-III Major 5 2026-27			
Hours	90 (60+ 30)	L	T	P	C
Pre requisites	Basic Computer Knowledge	3	-	1	4

Course Objective
1. Introduce core OOP principles and contrast procedural and object-oriented paradigms within the Java ecosystem. 2. Equip learners with foundational and advanced Java syntax including variables, control statements, arrays, strings, and classes. 3. Enable the use of inheritance, polymorphism, interfaces, and exception handling to create maintainable and reusable code. 4. Train students in concurrent programming and stream-based I/O operations including file management and serialization. 5. Empower learners to design, build, and manage GUI programs using Swing components, layout managers, and event handling techniques.

Course Outcomes

Outcome	Description	CognitiveLevel
C01	Apply OOP principles such as encapsulation, inheritance, and polymorphism in Java applications.	Knowledge
C02	Write, compile, and debug Java code using control statements, arrays, classes, and methods effectively.	Knowledge
C03	Construct modular code leveraging interfaces, abstract classes, and package hierarchies	Analysis and Evaluation
C04	Manage thread lifecycles, synchronization, and I/O streams for file handling and console interaction	Application
C05	Design user interfaces using Swing and handle keyboard/mouse input through event-driven programming	Creativity

P R GOVT COLLEGE (AUTONOMOUS), KAKINADA
DEPARTMENT OF COMPUTER APPLICATIONS
II B.Sc (IT) Semester- III (W.E.F.2026-27)
OBJECT ORIENTED PROGRAMMING THROUGH JAVA
SYLLABUS

UNIT-I: OOPs Concepts and Java Programming:

Introduction to Object-Oriented concepts, procedural and object-oriented programming paradigm.

Java programming: An Overview of Java, Java Environment, Data types, Variables, constants, scope and life time of variables, operators, type conversion and casting, Accepting Input from the Keyboard, Reading Input with Java.util.Scanner Class, Displaying Output, Displaying Formatted Output with String.format(), Control Statements

UNIT-II : Arrays and OOP Constructs:

Arrays: Arrays, Command Line Arguments, Strings-String Class Methods

Classes & Objects: Creating Classes, declaring objects, Methods, parameter passing, static fields and methods, Constructors, and 'this' keyword, overloading methods and access

Inheritance: Inheritance hierarchies, super and subclasses, member access rules, 'super' keyword, preventing inheritance: final classes and methods, the object class and its methods; **Polymorphism:** Dynamic binding, method overriding, abstract classes and methods;

UNIT-III : Interfaces, Packages & Exception Handling:

Interfaces VS Abstract classes, defining an interface, implement interfaces, accessing implementations through interface references, extending interface

Packages: Defining, creating and accessing a package, importing packages.

Exception Handling: Benefits of exception handling, the classification of exceptions, exception hierarchy, checked exceptions and unchecked exceptions, usage of try, catch, throw, throws and finally, rethrowing exceptions, exception specification, built in exceptions, creating own exception sub classes.

UNIT-IV : Multithreading & Stream based I/O:

Differences between multiple processes and multiple threads, thread states, thread life cycle, creating threads, interrupting threads, thread priorities, synchronizing threads, inter thread communication.

Stream based I/O (java.io) – The Stream classes-Byte streams and Character streams, Reading console Input and Writing Console Output, File class, Reading and writing Files, The Console class, Serialization

UNIT- V : GUI Programming with Swing:

Introduction, MVC architecture, components, containers. Understanding Layout Managers - Flow Layout, Border Layout, Grid Layout, Card Layout, GridBag Layout. Event Handling- The Delegation event model- Events, Event sources, Event Listeners, Event classes, Handling mouse and keyboard events, Adapter classes.

Text Books:

1. Java The complete reference, Herbert Schildt, 9th edition, McGraw Hill.
2. Programming in Java, S. Malhotra, S. Chudhary, 2nd edition, Oxford Univ. Press.

Reference Books:

1. Programming with JAVA - A Primer, E Balaguruswamy, 3rd Edition, McGraw Hill
2. Head First Java: A Brain-Friendly Guide , Katty Sierra, Bert Bates, 2nd Edition, O'Reilly

Activities:

Outcome: Apply OOP principles such as encapsulation, inheritance, and polymorphism in Java applications

Activity: Develop a class hierarchy for a zoo management system using inheritance and polymorphism (e.g., Animal → Mammal → Dog). Implement encapsulation through private fields and public getters/setters.

Evaluation Method: Code review and oral explanation focusing on class relationships, method overriding, and encapsulation practices.

Outcome: Write, compile, and debug Java code using control statements, arrays, classes, and methods effectively

Activity: Create a console-based student grade calculator using loops, conditionals, arrays, and modular methods.

Evaluation Method: Practical test with debugging tasks and output validation across multiple input scenarios.

Outcome: Construct modular code leveraging interfaces, abstract classes, and package hierarchies

Activity: Design a payment processing system with abstract classes for Payment, interfaces for Taxable, and organize classes into packages (e.g., com.billing, com.tax).

Evaluation Method:

Project submission assessed for modularity, interface implementation, abstraction usage, and package structure.

Outcome: Manage thread lifecycles, synchronization, and I/O streams for file handling and console interaction

Activity: Build a multithreaded logger that reads input from the console and writes to a file using synchronized threads and buffered streams.

Evaluation Method: Lab demonstration with thread state tracing and file output verification under concurrent input.

Outcome: Design user interfaces using Swing and handle keyboard/mouse input through event-driven programming

Activity: Create a GUI-based quiz application using Swing components (JFrame, JButton, JTextField) with event listeners for mouse clicks and key presses.

Evaluation Method: Live demo and rubric-based assessment of UI responsiveness, event handling accuracy, and layout design.

PRACTICAL**List of Experiments**

1. Write a Java program to print Fibonacci series.
2. Write a Java program to calculate multiplication of 2 matrices.
3. Write a Java program for sorting a given list of names in ascending order.
4. Create a class Rectangle. The class has attributes length and width. It should have methods that calculate the perimeter and area of the rectangle. It should have readAttributes() method to read length and width from the user.
5. Write a Java program that implements method overloading.
6. Write a Java program to implement various types of inheritance
 - i. Single ii. Multi-Level iii. Hierarchical iv. Hybrid
7. Write a java program to implement runtime polymorphism.
8. Write a Java program which accepts withdrawal amount from the user and throws an exception In Sufficient Funds when withdrawal amount is more than available amount.
9. Write a Java program to create three threads and that displays good morning, for every one second, hello for every 2 seconds and welcome for every 3 seconds by using extending Thread class.
10. Write a Java program that creates three threads. The first thread displays OOPS, the second thread displays Through and the third thread displays JAVA by using Runnable interface.
11. Write a Java program that displays the number of characters, lines and words in a text file.
12. Implement a Java program for handling mouse events when the mouse entered, exited, clicked, pressed, released, dragged and moved in the client area.
13. Implement a Java program for handling key events when the key board is pressed, released, typed.
14. Write a Java swing program that reads two numbers from two separate text fields and displays the sum of two numbers in the third text field when the button add is pressed.
15. Write a Java program to design student registration form using Swing Controls. The form which having the following fields and button SAVE

Form Fields are: Name, RNO, Mailid, Gender, Branch, Address.

P.R.GOV.T.COLLEGE (AUTONOMOUS), KAKINADA
DEPARTMENT OF COMPUTER APPLICATIONS
II B.Sc- IT SEMESTER-III (W.E.F.2026-27)

SUBJECT: OOP'S THROUGH JAVA

Time:2Hrs
Marks: 50

Model blue print for the model paper and choice

S.NO	Type of Question	To be given in the Question Paper			To be answered		
		No. of Questions	Marks allotted to each question	Total Marks	No. of Questions	Marks allotted to each question	Total Marks
1	Section-I Essay Questions	6	10	60	3	10	30
2	Section-II Short Questions	7	5	35	4	5	20
TOTAL		13		95	TOTAL MARKS		50

Model Blue print for the question paper Setter

Chapter Name	Essay Questions 10 Marks	Short Questions 5 Marks	Marks allotted to the chapter
UNIT-I	2	2	30
UNIT-II	1	2	20
UNIT-III	1	1	15
UNIT-IV	1	1	15
UNIT-V	1	1	15
Total No.of questions	6	7	
Total Marks Including choice			95

P R GOVT COLLEGE (AUTONOMOUS), KAKINADA
DEPARTMENT OF COMPUTER APPLICATIONS
II B.Sc (IT) Semester- III (2026-27)
OOP'S THROUGH JAVA
MODEL PAPER

Time: 2Hrs

Max.Marks: 50

SECTION-I

Answer Any Three Questions. At least One question from each part (3x10=30M)

Part-A

1. Explain the features of Java?.
2. Explain different Loop statements available in Java.
3. What is Inheritance? Explain Different types of Inheritance in Java.

Part-B

4. What is a package? Explain the process of creating and using packages?
5. Explain how exception handling is done in java.
6. How to work with Event Handling in Java?

SECTION-II

Answer any FOUR Questions. Each question carries 5 marks (4x5=20M)

7. What are the data types supported by Java?
8. What is the difference between break and continue statements in java.
9. List and explain any five String class methods in java
10. Explain about access specifiers in java?
11. What is an Interface?
12. Explain about thread priority?
13. Describe Grid Layout.

PR GOVERNMENT COLLEGE (AUTONOMOUS), KAKINADA					
Course Code: 25IT32	DATA BASE MANAGEMENT SYSTEM	II BSC (Major-6) Sem - III 2026-2027			
Hours	90 (60 Theory + 30 Practical)	L	T	P	C
Pre requisites	Basic Computer Knowledge	3	-	1	4

Course Objectives:

1. To understand the fundamentals of data, information, and the evolution from file-based systems to modern database management systems.
2. To develop the ability to design conceptual data models using Entity-Relationship (ER) and Enhanced ER diagrams.
3. To explore relational model principles, such as keys, integrity constraints, relational algebra and calculus, and normalization.
4. To perform data definition and manipulation using SQL commands including queries, joins, subqueries, views, and set operations.
5. To apply procedural logic using PL/SQL, incorporating control structures, functions, procedures, and database.

Course Outcomes	
On Completion of the course, the students will be able to –	
C01	Describe the fundamentals of data, database systems, and the differences between file-based and database approaches. Compare and classify various DBMS architectures, data models, and their components, including the three-schema architecture.
C02	Design conceptual data models using Entity-Relationship and Enhanced ER diagrams, applying generalization, specialization, and constraints.
C03	Apply relational model concepts, including CODD rules, relational algebra, relational calculus, and normalization techniques.
C04	Construct and execute SQL queries for data definition, manipulation, aggregation, joining, and subqueries, including views and set operations.
C05	Develop PL/SQL programs incorporating control structures, procedures, and functions to manage database behavior effectively.

P R GOVT COLLEGE(AUTONOMOUS), KAKINADA
DEPARTMENT OF COMPUTER APPLICATIONS
II BSC (Major) Semester- III (2026-2027)
Data Base Management System
SYLLABUS

Unit 1. Overview of Database Management System:

Introduction to data, information, database, database management systems, file-based system, Drawbacks of file-Based System, database approach, Classification of Database Management Systems, advantages of database approach, Various Data Models, Components of Database Management System, three schema architecture of data base, costs and risks of database approach.

Unit 2. Entity-Relationship Model:

Introduction, the building blocks of an entity relationship diagram, classification of entity sets, attribute classification, relationship degree, relationship classification, reducing ER diagram to tables, enhanced entity-relationship model (EER model), generalization and specialization, IS A relationship and attribute inheritance, multiple inheritance, constraints on specialization and generalization, advantages of ER modeling.

Unit 3. Relational Model:

Introduction, CODD Rules, relational data model, concept of key, relational integrity, relational algebra, relational algebra operations, advantages of relational algebra, limitations of relational algebra, Functional dependencies and normal forms.

Unit 4. Structured Query Language:

Introduction, Commands in SQL, Data Types in SQL, Data Definition Language, Selection Operation, Projection Operation, Aggregate functions, Data Manipulation Language, Table Modification Commands, Join Operation, Set Operations, View, Sub Query.

Unit 5. PL/SQL:

Introduction, Shortcomings of SQL, Structure of PL/SQL, PL/SQL Language Elements, Data Types, Operators Precedence, Control Structures, Steps to Create a PL/SQL, Program, Iterative Control, Procedures, Functions.

Textbooks:

1. Database System Concepts, Avi Silberschatz, Henry F. Korth, S. Sudarshan, Seventh Edition, McGraw-Hill
2. Database Management Systems by Raghu Ramakrishnan, McGrawhill

Reference Books:

1. Fundamentals of Database Systems, Elmasri Navathe Pearson Education
2. An Introduction to Database systems, C.J. Date, A.Kannan, S.Swami Nadhan, Pearson

Activities:

Outcome: Describe the fundamentals of data, database systems, and the differences between file-based and database approaches. Compare and classify various DBMS architectures, data models, and their components, including the three-schema architecture.

Activity: Create a comparative presentation or infographic illustrating:

- File-based vs. DBMS approaches
- Types of DBMS architectures (1-tier, 2-tier, 3-tier)
- Data models and the three-schema architecture

Evaluation Method: Rubric-based assessment of the presentation covering clarity, accuracy, and depth of comparison. Include a short quiz to test conceptual understanding.

Outcome: Design conceptual data models using Entity-Relationship and Enhanced ER diagrams, applying generalization, specialization, and constraints.

Activity: Model a university or hospital database using ER and Enhanced ER diagrams that shows:

- Entity sets, relationships
- Generalization/specialization
- Participation and cardinality constraints

Evaluation Method: Diagram submission with peer review and instructor feedback. Use a checklist to assess completeness, correctness, and notation usage.

Outcome: Apply relational model concepts, including CODD rules, relational algebra, relational calculus, and normalization techniques.

Activity: Normalize a given unstructured dataset up to 3NF. Then, write relational algebra expressions for sample queries.

Evaluation Method: Written assignment graded on:

- Correctness of normalization steps
- Accuracy of relational algebra expressions
- Short-answer questions on CODD rules and relational calculus

Outcome: Construct and execute SQL queries for data definition, manipulation, aggregation, joining, and subqueries, including views and set operations.

Activity: Implement a mini-project (e.g., Library or Inventory DB) using SQL. Include:

- Table creation (DDL)
- Data manipulation (DML)
- Aggregation, joins, subqueries, views, and set operations

Evaluation Method: Lab-based practical test with query execution and output validation. Include a viva to explain logic and optimization.

Outcome: Develop PL/SQL programs incorporating control structures, procedures and functions to manage database behaviour effectively.

Activity: Build a PL/SQL-based payroll or student grading system using:

- Procedures and functions
- Control structures (IF, LOOP)

COURSE 2: DATABASE MANAGEMENT SYSTEMS

Practical

Credits: 1

2 hrs/week

Experiment 1 : Database: Inventory Management

Table 1: Products

Structure:

Column Name	Data Type	Constraints
product_id	INT	PRIMARY KEY
product_name	VARCHAR(50)	NOT NULL
price	DECIMAL(10,2)	CHECK(price > 0)
stock_qty	INT	CHECK(stock_qty >= 0)

Sample Data:

product_id	product_name	price	stock_qty
1	Pen	10.00	100
2	Notebook	50.00	200
3	Stapler	120.00	50
4	Marker	25.00	80
5	File Folder	60.00	150

Table 2:

Suppliers

Structure

Column Name	Data Type	Constraints
supplier_id	INT	PRIMARY KEY
supplier_name	VARCHAR(50)	NOT NULL
contact_no	VARCHAR(20)	UNIQUE
product_id	INT	FOREIGN KEY REFERENCES Products(product_id)

Sample Data:

supplier_id	supplier_name	contact_no	product_id
101	StationeryMart	9876543210	1
102	PaperWorld	9876500000	2
103	OfficeSupplies	9876512345	3
104	MarkerHub	9876522222	4
105	FileDepot	9876533333	5

Section A: DDL (Data Definition Language)

1. Create a database called InventoryDB.
2. Create a table Products and table Suppliers with the specified columns and constraints:

Section B: DML (Data Manipulation Language)

4. Insert at least 5 rows into the Products table.
5. Insert at least 5 rows into the Suppliers table.
6. Update the stock quantity of product 'Pen' to 120.
7. Delete a supplier with a specific supplier_id.
8. Write a query to rename 'Notebook' to 'NoteBook A4'

Section C: DQL (SELECT Queries)

9. **Display all records from the Products table.**
10. Display only product_name and price of all products.
11. List all products that have a stock quantity less than 100.
12. Show all products between 20 and 100 price range.
13. Find all suppliers whose contact number starts with '98765'.
14. Find the average price of products.
15. Display the total number of products in the inventory.
16. Show the maximum and minimum stock quantities.
17. Count how many suppliers supply each product.
18. Show all products where price > 50 AND stock_qty > 100.
19. Show all products where price < 20 OR stock_qty < 80.
20. Display suppliers whose supplier_name contains the word 'Mart'
21. List all suppliers along with the product they supply (use INNER JOIN).
22. Display suppliers whose name starts with 'S'.
23. Find products whose name has exactly 5 characters
24. Find suppliers who supply products costing more than 100.

Experiment 2 : ONLINE BOOKSTORE DB

An online book store wants to implement a BOOKSTORE DB for managing their online transactions by using the following tables.

Authors Table

Column Name	Data Type	Constraints
author_id	INTEGER	PRIMARY KEY
first_name	VARCHAR	NOT NULL
last_name	VARCHAR	NOT NULL
nationality	VARCHAR	NULL allowed

Books Table

Column Name	Data Type	Constraints
book_id	INTEGER	PRIMARY KEY
Title	VARCHAR	NOT NULL
author_id	INTEGER	FOREIGN KEY REFERENCES Authors
publication_year	INTEGER	
Price	DECIMAL	

Customers Table

Column Name	Data Type	Constraints
customer_id	INTEGER	PRIMARY KEY
first_name	VARCHAR	NOT NULL
last_name	VARCHAR	NOT NULL
Email	VARCHAR	UNIQUE, NOT NULL
Address	VARCHAR	NOT NULL

Orders Table

Column Name	Data Type	Constraints
order_id	INTEGER	PRIMARY KEY
customer_id	INTEGER	FOREIGN KEY REFERENCES Customers
book_id	INTEGER	FOREIGN KEY REFERENCES Books
order_date	DATE	NOT NULL
quantity	INTEGER	NOT NULL

SAMPLE DATA SET for BOOKSTORE DB

Authors Table

author_id	first_name	last_name	nationality
1	Jane	Austen	British
2	George	Orwell	British
3	Gabriel	Garcia Marquez	Colombian
4	Toni	Morrison	American
5	Mark	Twain	American
6	Harper	Lee	American
7	Fyodor	Dostoevsky	Russian

Books Table

book_id	Title	author_id	publication_year	price
101	Pride and Prejudice	1	1813	12.99
102	1984	2	1949	9.50
103	One Hundred Years of Solitude	3	1967	15.00
104	Beloved	4	1987	11.25
105	Animal Farm	2	1945	8.75
106	Adventures of Huckleberry Finn	5	1884	10.50
107	To Kill a Mockingbird	6	1960	14.00

Customers Table

customer_id	first_name	last_name	Email	address
201	Alice	Smith	alice.s@example.com	12 Oak St, London
202	Bob	Johnson	bob.j@example.com	45 Pine Ave, Oxford
203	Charlie	Brown	charlie.b@example.com	78 Maple Rd, Bristol
204	Diana	Prince	diana.p@example.com	34 Queen St, York
205	Edward	Norton	edward.n@example.com	22 River Ln, Leeds
206	Fiona	Hall	fiona.h@example.com	56 Lake Dr, Bath
207	Greg	Miller	greg.m@example.com	89 Park Ave, Glasgow

Orders Table

order_id	customer_id	book_id	order_date	Quantity
301	201	101	2025-07-20	1
302	202	102	2025-07-21	2
303	201	105	2025-07-22	1
304	203	103	2025-07-23	1
305	204	106	2025-07-24	1
306	205	107	2025-07-25	3
307	206	104	2025-07-26	2

Section A: DDL (Schema Design & Constraints)

1. Write SQL statements to create all 4 tables (Authors, Books, Customers, Orders) with:
 - o Primary Keys
 - o Foreign Keys
 - o Appropriate data types
 - o NOT NULL constraints where necessary.
2. Alter the Books table to add a constraint that price must be greater than 0.
3. Add a new column phone_number to the Customers table (VARCHAR(15)) and ensure it is unique.
4. Drop the phone_number column from the Customers table.

Section B: DML (Data Manipulation)

5. Insert at least 7 records for each table (use sample dataset above).
6. Update the price of the book titled *Animal Farm* by increasing it by 10%.
7. Delete all orders made before 2025-07-21.
8. Change the nationality of Gabriel Garcia Marquez to “Latino-American”.

Section C: SELECT Queries (Data Querying)

9. List all books published between 1900 and 2000.
10. Find all customers whose email contains “example.com”.
11. Retrieve books whose price is between 10 and 15 and published before 1950.
12. Show authors who are either ‘British’ or ‘American’.
13. Find books that have a price less than 10 or are published after 1980.
14. Display all orders placed after 2025-07-22.
15. List all books written by author with author_id = 2.
16. Find customers whose last name starts with B.
17. Show all books with a price NOT between 9 and 13.
18. Display books whose publication_year is in (1813, 1945, 1987).
19. Find authors whose nationality is NOT ‘British’.
20. List customers whose address contains the word Park.
21. Show all books sorted by price in descending order.
22. List authors in alphabetical order by last_name.
23. Display orders sorted by order_date (latest first). Use of Date Functions
24. Show all orders placed in July 2025.
25. Show all orders with an estimated delivery date (5 days after order date).
26. Show customers who placed an order on a weekend.
27. Calculate how many days have passed since the last order was placed.

Aggregate Functions (COUNT, SUM, AVG, MIN, MAX)

28. Count the total number of books in the database.
29. Find the average price of all books.
30. Show the highest-priced book.
31. Count how many orders each customer has placed.
32. Calculate the total sales (price × quantity) for each customer.

GROUP BY and HAVING

33. Count how many books are written by each author.
34. Group orders by customer_id and display total quantity ordered.
35. Show customers who have ordered more than 2 books in total (use HAVING).
36. Find the total number of books sold per author (GROUP BY author).

Experiment 3: EMPLOYEE DB

An enterprise wants to automate its employee management process by implementing an Employee Database. The goal is to replace manual record-keeping with a centralized system that stores employee, department, and project details. Use the following table structures and data set to implement Employee DB.

Employee DB – Table Structures

1. Departments Table

Column	Type	Constraints
dept_id	INT	PRIMARY KEY
dept_name	VARCHAR	UNIQUE, NOT NULL
location	VARCHAR	NOT NULL

2. Employees Table

Column	Type	Constraints
emp_id	INT	PRIMARY KEY
first_name	VARCHAR	NOT NULL
last_name	VARCHAR	NOT NULL
email	VARCHAR	UNIQUE, NOT NULL
phone	VARCHAR	CHECK (phone LIKE '--____')
hire_date	DATE	NOT NULL
job_title	VARCHAR	NOT NULL
salary	DECIMAL	CHECK (salary > 0)
dept_id	INT	FOREIGN KEY REFERENCES Departments(dept_id)
manager_id	INT	FOREIGN KEY REFERENCES Employees(emp_id) (self-referential)

3. Projects Table

Column	Type	Constraints
project_id	INT	PRIMARY KEY
project_name	VARCHAR	NOT NULL
start_date	DATE	NOT NULL
end_date	DATE	NULL
dept_id	INT	FOREIGN KEY REFERENCES Departments(dept_id)

1. Employee_Project Table (Many-to-Many)

Column	Type	Constraints
emp_id	INT	FOREIGN KEY REFERENCES Employees(emp_id), PRIMARY KEY(emp_id, project_id)
project_id	INT	FOREIGN KEY REFERENCES Projects(project_id)
hours_allocated	INT	CHECK (hours_allocated > 0)

Sample Data Set

Departments

dept_id	dept_name	Location
1	HR	New York
2	IT	San Francisco
3	Finance	Chicago
4	Marketing	Boston
5	Operations	Seattle
6	Legal	Washington D.C.
7	Sales	Dallas
8	R&D	Austin
9	Procurement	Denver
10	Customer Care	Miami

2. Employees Table

emp_id	first_name	last_name	Email	phone	hire_date	job_title	salary	dept_id	manager_id
101	Alice	Johnson	alice.j@corp.com	123-456-7890	2020-03-15	HR Manager	75000	1	NULL
102	Bob	Smith	bob.s@corp.com	234-567-8901	2019-05-20	IT Analyst	65000	2	104
103	Charlie	Brown	charlie.b@corp.com	345-678-9012	2021-01-10	Finance Executive	58000	3	106
104	Diana	Prince	diana.p@corp.com	456-789-0123	2018-07-12	IT Manager	90000	2	NULL
105	Ethan	Hunt	ethan.h@corp.com	567-890-1234	2022-02-25	Marketing Lead	62000	4	NULL
106	Fiona	Hall	fiona.h@corp.com	678-901-2345	2017-11-01	Finance Manager	85000	3	NULL

107	Greg	Miles	greg.m@corp.com	789-012-3456	2023-04-15	IT Support	45000	2	104
108	Hannah	White	hannah.w@corp.com	890-123-4567	2021-09-05	HR Executive	50000	1	101
109	Ian	Scott	ian.s@corp.com	901-234-5678	2020-11-20	Operations Analyst	56000	5	NULL
110	Julia	Adams	julia.a@corp.com	012-345-6789	2019-12-18	Legal Advisor	70000	6	NULL

2. Projects Table

project_id	project_name	start_date	end_date	dept_id
201	Payroll System	2023-01-01	NULL	3
202	Website Upgrade	2023-02-10	NULL	2
203	Recruitment Drive	2023-03-05	NULL	1
204	Ad Campaign	2023-05-20	NULL	4
205	New CRM Tool	2023-04-15	NULL	7
206	Compliance Portal	2023-06-10	NULL	6
207	Inventory System	2023-07-01	NULL	5
208	AI Research	2023-08-05	NULL	8
209	Customer Feedback	2023-09-10	NULL	10
210	Procurement System	2023-10-01	NULL	9

3. Employee_Project Table

emp_id	project_id	hours_allocated
102	202	120
104	202	80
103	201	100
106	201	150
101	203	50
105	204	70
107	202	60
109	207	90
110	206	110
108	203	40

Section A: DDL (Schema Creation & Modification)

1. Write SQL statements to create the above tables with the specified constraints
2. Alter the Employees table to add a column bonus DECIMAL(8,2) with default value 0.

3. Drop the column bonus from Employees.

Section B: DML (Insert, Update, Delete)

4. Insert at least 10 rows into Departments, Employees, Projects, and Employee_Project.(use the above data set)
5. Try inserting an employee with a negative salary (should fail due to CHECK constraint).
6. Update the salary of the employee with emp_id = 103 by 15%.
7. Delete an employee record who has resigned (choose any emp_id).
8. Increase all employees' salaries in the IT department by 5%.
9. Change the department of an employee to "Research".(should fail due to FK constraint)

Section C: DQL (Select Queries)

10. List all employees and their details.
11. Show all employees in the "HR" department.
12. Find employees with salaries between 50,000 and 80,000.
13. Retrieve employees hired after 2020.
14. Show employees who are in either the IT or Finance department.
15. Find employees whose email ends with "@corp.com".
16. List all employees with salary > 60,000 AND located in "New York".
17. Display employees in descending order of salary.
18. Count the number of employees in each department.
19. Show the average salary of employees department-wise.

Section D: Joins

20. List all employees with their department names (INNER JOIN).
21. Display all departments along with employees, including those departments without employees (LEFT JOIN).
22. Show employees and the projects they are working on (JOIN 3 tables: Employees, Employee_Project, Projects).

23. List projects along with total hours allocated by employees.
24. Write a query to find employees who are working on more than one project.
25. Show all projects handled by the 'Finance' department.

Section E: PL/SQL Programming

1. Write a procedure GetEmpInfo that takes emp_id as input and displays name, salary, and department.
2. Write a PL/SQL block that checks if an employee's salary is above 50,000. If yes, print "High Salary" ;Otherwise print "Standard Salary".
3. Write a PL/SQL program to display the top 10 rows in the Emp table based on their job and salary
4. Write a stored procedure GiveBonus that takes department ID and a designation as input, along with a bonus amount, and updates the salary of all employees in that department who have the specified designation by adding the bonus amount to their current salary.
5. Create a trigger to prevent inserting employees with a salary less than 30,000.
6. Create a trigger to avoid any transactions (insert, update, delete) on EMP table on Saturday & Sunday.

P R GOVT COLLEGE(AUTONOMOUS), KAKINADA
DEPARTMENT OF COMPUTER APPLICATIONS
II BSC (Major) Semester-III (W.E.F.2026-2027)
Data Base Management System

Marks: 50M

Model blue print for the model paper and choice

S.NO	Type of Question	To be given in the Question Paper			To be answered		
		No. of Questions	Marks allotted to each question	Total Marks	No. of Questions	Marks allotted to each question	Total Marks
1	Section-A Essay Questions	6	10	60	3	10	30
2	Section-B Short Questions	7	5	35	4	5	20
TOTAL		13		95	TOTAL MARKS		50

Model Blue print for the question paper setter

Chapter Name	Essay Questions 10 Marks	Short Questions 5 Marks	Marks allotted to the chapter
UNIT-I	2	2	30
UNIT -II	1	1	15
UNIT -III	1	1	15
UNIT -IV	1	2	20
UNIT -V	1	1	15
Total No. of questions	6	7	
Total Marks Including choice			95

P R GOVT COLLEGE(AUTONOMOUS), KAKINADA
DEPARTMENT OF COMPUTER APPLICATIONS
II BSC (Major) Semester-IV (W.E.F.2026-2027)
MODEL PAPER

Database Management System

Time : 2 Hrs

SEMESTER-III

Max. Marks: 50

SECTION-I

Answer Any Three Questions. At least One question from each part

(3x10=30M)

Part-A

1. What is meant by DBMS? Explain advantages of DBMS.(BT1)
2. Explain about characteristics and drawbacks of File based system(BT2)
3. Discuss about building blocks of Entity-Relationship diagram(BT2)

Part-B

4. What is data model? Write about relational data model (BT1)
5. Explain about SQL commands. (BT2)
6. What is PL/SQL? Write about structure of PL/SQL with example.(BT1)

SECTION-II

Answer any FOUR Questions. Each question carries 5 marks

(4x5=20M)

7. Explain about drawbacks of file-based system. (BT2)
8. Distinguish between data and information. (BT3)
9. Explain about entity sets. (BT2)
10. Explain about i) Candidate key ii) Primary key iii) Foreign key (BT2)
11. Write about datatypes in sql (BT1)
12. Explain about selection and projection operators. (BT1)
13. Write about cursors in PL/SQL (BT1)

PR GOVERNMENT COLLEGE (AUTONOMOUS), KAKINADA					
Course Code: 25IT33	COMPUTER ORGANIZATION	II B.Sc- (IT) Major 7 Sem - III 2026-27			
Hours	90 (60 Theory + 30 Practical)	L	T	P	C
Pre requisites		3	-	2	4

Course Objective
The objective of this course is to provide students 1.To understand the purpose of registers, register transfer language and instructions 2.To understand the Memory, addressing modes and Central processing unit 3.To understand the peripheral devices 4.To understand arithmetic operations through algorithms

Course Outcomes	
Students after successful completion of the course will be able to:	
CO1	1. Identify different types of instructions
CO2	2. Differentiate between micro-programmed and hard-wired control units.
CO3	3. Analyse the performance of hierarchical organization of memory.
CO4	4. Summarize different data transfer techniques.
CO5	5. Demonstrate arithmetic operations on fixed- and floating-point numbers and illustrate concepts of parallel processing.

P R GOVT COLLEGE(AUTONOMOUS), KAKINADA
DEPARTMENT OF COMPUTER APPLICATIONS
II B.Sc (IT) Semester- III(2026-27)
COMPUTER ORGANIZATION
SYLLABUS.

UNIT I

Register Transfer Language and Micro Operations: Introduction- Functional units, computer registers, register transfer language, register transfer, bus and memory transfers, arithmetic, logic and shift micro-operations, arithmetic logic shift unit. Basic Computer Organization and Design: Instruction codes, instruction cycle. Register reference instructions, Memory – reference instructions, input – output and interrupt.

UNIT II

CPU and Micro Programmed Control: Central Processing unit: Introduction, instruction formats, addressing modes. Control memory, address sequencing, design of control unit - hard wired control, micro programmed control.

UNIT III

Memory Organization: Memory hierarchy, main memory, auxiliary memory, associative memory, cache Memory and mappings.

UNIT IV

Input-Output Organization: Peripheral Devices, input-output interface, asynchronous data transfer, modes of transfer- programmed I/O, priority interrupt, direct memory access, Input – Output Processor (IOP).

UNIT V

Computer Arithmetic: Data representation- fixed point, floating point, addition and subtraction, multiplication and division algorithms.

REFERENCES TEXT BOOK:

1. Computer Systems Architecture, M. Moris Mano, 3rd edition, Pearson/ PHI
2. Computer Organization and Architecture, William Stallings, 8th edition, Pearson/PHI

REFERENCES

1. Computer Organization – Carl Hamacher, Zvonko G- Vranesic, Safwat G, Zaky Fifth Edition, Mc- Grawhill INC.,
2. Rajaraman& T. Radhakrishnan: Computer Organization and Architecture. PHI
3. Donald D. Givone: Digital Principles and Design. McGraw Hill

SUGGESTED CO-CURRICULAR ACTIVITIES:

1. Assignments
2. Seminars, Group discussions, Quiz, Debates etc. (on related topics).

3. Presentation by students on applications related to Graph Theory

COMPUTER ARCHITECTURE – PRATICAL

List of Experiments:

1. Simulate Register Transfer Using Logisim
2. Build a microprogrammed control unit using Logisim or VHDL (GHDL).
3. Simple Register Transfers and Arithmetic Operations using EMU8086 Simulator
4. Simulate a Simple Instruction Cycle in Emu8086
5. Demonstrate Addressing Modes (Immediate, Register, Direct, Indexed) in Emu8086
6. Implement Subroutines Using CALL and RET in Emu8086
7. Simulate DMA Transfer in Ripes or Logisim
8. Simulate Booth's Algorithm for multiplication and restoring division in Python/C or Logisim
9. Write assembly language code for $A+B*(C-D)$ using various instruction formats in any open-source assembler.
10. Write assembly language code for $A+B*C$ using various addressing modes in any open-source assembler.
11. Implement a C program to perform Binary Addition & Subtraction.
12. Implement a C program to perform Multiplication of two binary numbers.
13. Implement a C program to perform Multiplication of two binary numbers (signed) using Booth's Algorithms.
14. Implement a C program to perform division of two binary numbers (Unsigned) using restoring division algorithm.

P R GOVT COLLEGE(AUTONOMOUS), KAKINADA
DEPARTMENT OF COMPUTER APPLICATIONS
II B.Sc (IT) Semester- III (2026-27)
COMPUTER ORGANIZATION

Marks: 50M

Model blue print for the model paper and choice

S.NO	Type of Question	To be given in the Question Paper			To be answered		
		No. of Questions	Marks allotted to each question	Total Marks	No. of Questions	Marks allotted to each question	Total Marks
1	Section-A Essay Questions	6	10	60	3	10	30
2	Section-B Short Questions	7	5	35	4	5	20
TOTAL		13		95	TOTAL MARKS		50

$$\text{Percentage of choice given} = \frac{95 - 50}{95} \times 100 = \frac{45}{95} \times 100 = 47.36\%$$

Model Blue print for the question paper setter

Chapter Name	Essay Questions 10 Marks	Short Questions 5 Marks	Marks allotted to the chapter
UNIT-I	2	2	30
UNIT -II	1	2	20
UNIT -III	1	1	15
UNIT -IV	1	1	15
UNIT -V	1	1	15
Total No. of questions	6	7	
Total Marks Including choice			95

P R GOVT COLLEGE(AUTONOMOUS), KAKINADA
DEPARTMENT OF COMPUTER APPLICATIONS
II B.Sc (IT) Semester- III (2026-27)
COMPUTER ORGANIZATION
MODEL PAPER

Time : 2Hrs

Max. Marks: 50

SECTION-I

Answer Any Three Questions. At least One question from each part (3x10=30M)

Part-A

1. Explain the concept of functional units in computer organization
2. What are addressing modes in the design of a CPU?
3. What is Direct Memory Access (DMA) and how does it improve data transfer efficiency?

Part-B

4. How are bus and memory transfers crucial in computer organization?
5. What is the purpose of a memory hierarchy in a computer system?
6. Describe the multiplication algorithm for fixed point numbers.

SECTION-II

Answer any FOUR Questions. Each question carries 5 marks (4x5=20M)

7. What is the role of computer registers in a CPU?
8. Describe the stages involved in the instruction cycle of a CPU.
9. How do different instruction formats affect the operation of a CPU?
10. How is control memory utilized in a CPU?
11. How does main memory differ from auxiliary memory?
12. How does a priority interrupt work in a computer system?
13. Differentiate between arithmetic and instruction pipeline in parallel processing

II B.Sc (IT) (W.E.F 2026-2027)
SEMESTER – III
QUESTION BANK

Subject: Computer Organization

Paper: 7

MODULE-I

Essay Questions:

1. Explain the concept of functional units in computer organization
2. Define register transfer language (RTL) and its significance in computer design.
3. How are bus and memory transfers crucial in computer organization?
4. What are arithmetic, logic, and shift micro-operations in CPU execution?

Short Questions:

5. What is the role of computer registers in a CPU?
6. How do instruction codes impact basic computer organization and design?
7. Describe the stages involved in the instruction cycle of a CPU.
8. What distinguishes register reference instructions from memory-reference instructions?

MODULE-II

Essay Questions:

1. What is the purpose of the Central Processing Unit (CPU) in a computer system?
2. What are addressing modes in the design of a CPU?
3. What is address sequencing, and why is it important in CPU design?

Short Questions:

4. How do different instruction formats affect the operation of a CPU?
5. How is control memory utilized in a CPU?
6. How do addressing modes and instruction formats interact in determining the performance of CPU?

MODULE-III

Essay Questions:

1. What is the purpose of a memory hierarchy in a computer system?
2. What is associative memory in the context of computer systems?
3. What are the different types of memory mappings used in computer systems?

Short Questions:

4. Why is it important to consider the size and associativity of cache memory in system design?

5. What role does virtual memory play in modern computer systems?
6. How does the use of memory hierarchy contribute to the overall efficiency of a computer system?
7. How does main memory differ from auxiliary memory?

MODULE-IV

Essay Questions:

1. What is the purpose of Peripheral Devices in a computer system?
2. What is asynchronous data transfer? Explain in-detail.
3. What is Direct Memory Access (DMA) and how does it improve data transfer efficiency?

Short Questions:

4. Explain the concept of input-output interface in the context of computer systems.
5. How does a priority interrupt work in a computer system?
6. How does the mode of transfer affect the efficiency of data transfer in a computer system?

MODULE-V

Essay Questions:

1. Describe the multiplication algorithm for fixed point numbers.
2. What is the division algorithm for floating point numbers?
3. Explain the concept of pipelining in parallel processing.

Short Questions:

4. Differentiate between arithmetic and instruction pipeline in parallel processing.
5. How does parallel processing impact the speed and efficiency of data processing tasks?
6. Explain the difference between fixed point and floating point data representation.

Mapping of Cos with POs/PSOs

[illegible]

SEMESTER-4

PR GOVERNMENT COLLEGE (AUTONOMOUS), KAKINADA					
Course Code 24IT41	OPERATING SYSTEM	II- BSC(IT) (MAJOR-8) SEM - IV 2026-27			
Hours	90 (60 Theory + 30 Practical)	L	T	P	C
Pre requisites	BASIC COMPUTER KNOWLEDGE	3	-	1	4

Course Objective

1.Understand the evolution and core functions of operating systems, including resource management and system types.
2.Analyze process and thread management , focusing on system calls, kernel modes, scheduling algorithms, and threading models.
3.Evaluate process synchronization and deadlock handling , using classical problems and inter-process communication methods
4.Apply memory management techniques , including paging, segmentation, and virtual memory implementation.
5.Examine file, I/O, and device management strategies , along with basic OS-level security features.

Course Outcomes: On Completion of the course, the students will be able to-		Cognitive Domain
CO1	Explain the foundational principles and evolution of operating systems, including resource abstraction and core management functions.	Understanding
CO2	Classify and compare different types of operating systems, such as multiprogramming, batch, time-sharing, real-time, and personal device-based systems.	Application
CO3	Analyze process and thread management techniques, including processor modes, system calls, kernel functions, and scheduling algorithms.	Application
CO4	Evaluate process synchronization and deadlock handling approaches, applying classical concurrency solutions and inter-process communication methods.	Application
CO5	Apply memory, file, and I/O management strategies, incorporating allocation techniques, virtual memory models, disk scheduling, and OS-level security features.	Application

P R GOVT COLLEGE (AUTONOMOUS), KAKINADA
DEPARTMENT OF COMPUTER APPLICATIONS
II BSC Semester-IV (W.E.F. 2026-27)
OPERATING SYSTEM

Unit 1. Operating System Fundamentals:

Operating System Definition, History and Evolution of OS, Basic OS functions, Types of Operating Systems– Multiprogramming Systems, Batch Systems, Time Sharing Systems; Operating Systems for Personal Computers, Workstations and Hand-held Devices, Process Control & Real time Systems.

Unit 2. Process & Thread:

Processor and User Modes, Kernels, System Calls and System Programs, System View of the Process and Resources, Process Abstraction, Process Hierarchy, Threads, Threading Issues, Thread Libraries; Process Scheduling- Non-Preemptive and Preemptive Scheduling Algorithms.

Unit 3. Process Management:

Concurrent and Dependent Processes, Critical Section, Semaphores, Methods for Inter process Communication; Process Synchronization, Classical Process Synchronization Problems: Producer-Consumer, Reader-Writer.

Deadlock, Deadlock Characterization, Necessary and Sufficient Conditions for Deadlock, Deadlock Handling Approaches: Deadlock Prevention, Deadlock Avoidance and Deadlock Detection and Recovery.

Unit 4. Memory Management:

Physical and Virtual Address Space; Memory Allocation Strategies–Fixed and -Variable Partitions, Paging, Segmentation, Virtual Memory

Unit 5. File and I/O Management:

Directory Structure, File Operations, File Allocation Methods, Device Management, Pipes, Buffer, Shared Memory, Disk Scheduling algorithms.

Text Books:

1. Operating System Principles, Abraham Silberschatz, Peter Baer Galvin and Greg Gagne, 7th Edition, Wiley India Edition.
2. Operating Systems: Internals and Design Principles, Stallings , Pearson edition

Reference Books:

1. Operating Systems Design and Implementation, Andrew S. Tanenbaum, 3rd Edition, Pearson
2. A Text Book of Operating Systems, Singh, Kaur and Gupta, Khanna Publishers

Activities:

Outcome: Explain the foundational principles and evolution of operating systems, including resource abstraction and core management functions.

Activity: Timeline Creation & Concept Mapping - Students will collaboratively build a visual timeline showing the evolution of operating systems from early batch systems to modern distributed and mobile OS. They will also create a concept map illustrating core management functions (e.g., process, memory, file, device management).

Evaluation Method: Rubric-based assessment of timeline and concept map:

- Historical accuracy
- Inclusion of core OS functions
- Presentation and teamwork

Outcome: Analyze process and thread management, focusing on system calls, kernel modes, scheduling algorithms, and threading models.

Activity: Comparative Analysis Table & Case Study Discussion - Students will research and fill out a comparison table for OS types (multiprogramming, batch, time-sharing, real-time, personal device-based), focusing on architecture, scheduling, responsiveness, and use cases. Followed by a group discussion using real-world examples (e.g., RTOS in pacemakers vs. Android in smartphones).

Evaluation Method: Students will be evaluated on a 10-point scale based on

- Individual submission of comparison table
- Group participation score in discussion

Outcome: Analyze process and thread management techniques, including processor modes, system calls, kernel functions, and scheduling algorithms.

Activity: Simulation & Code Walkthrough - Students will simulate process scheduling using tools or pseudocode (e.g., Round Robin, Priority Scheduling). They will also analyze thread creation and system calls using a simple multithreaded program in Java.

Evaluation Method: Students will be assessed on a 10-point scale based on

- Scheduling algorithm simulation results
- Explanation of processor modes and system calls
- Code annotations for kernel functions

Outcome: Evaluate process synchronization and deadlock handling approaches, applying classical concurrency solutions and inter-process communication methods.

Activity: Role-play & Coding Challenge - Students will role-play classical problems (e.g., DiningPhilosophersProducerConsumer) to understand synchronization. Then, they'll implement semaphore or monitor-based solutions in code and simulate deadlock detection or avoidance.

Evaluation Method: Students will be evaluated for 10 marks based on

- Code review for correctness and use of synchronization primitives
- Peer evaluation of role-play clarity and engagement
- Written test with deadlock scenarios and solution strategies

Outcome: Apply memory, file, and I/O management strategies, incorporating allocation techniques, virtual memory models, disk scheduling, and OS-level security features.

Activity: Interactive Lab & Design Task - Students will use OS simulators to experiment with memory allocation (paging, segmentation), disk scheduling (FCFS, SSTF, SCAN), and file system operations.

Evaluation Method: Students will be evaluated for 10 points based on

- Correct use of allocation techniques
- Disk scheduling output analysis
- Virtual memory behaviour observation.

SEMESTER-IV
COURSE 8: OPERATING SYSTEMS
Practical Credits: 1

2 hrs/

List of Experiments:

1. Illustrate the LINUX commands
 - a) pwd
 - b) mkdir
 - c) rmdir
 - d) grep
 - e) chmod
 - f) ls
 - g) rm
 - h) cp
2. Write a program to calculate average waiting time and turn around time of each process using the following CPU Scheduling algorithm for the given process schedules.
 - a) FCFS
 - b) SJF
 - c) Priority
 - d) Round Robin
3. Simulate MVT and MFT memory management techniques
4. Write a program for Bankers Algorithm for Dead Lock Avoidance
5. Implement Bankers Algorithm Dead Lock Prevention.
6. Write a program to simulate Producer-Consumer problem.
7. Simulate all Page replacement algorithms.
 - a) FIFO
 - b) LRU
 - c) LFU
 - d) Optimal
8. Simulate Paging Techniques of memory management.
9. Simulate the following disk scheduling algorithms.
 - a) FCFS
 - b) SSTF
 - c) SCAN
 - d) CSCAN

P. R.GOV'T. COLLEGE (AUTONOMOUS), KAKINADA
MODEL BLUE PRINT FOR THE YEAR 2026-2027
II B.SC (IT) 2026-2027BATCH
SEMESTER-IV
SUBJECT:OPERATING SYSTEMS

Time: 2 Hrs

Max. Marks: 50

Model blue print for the model paper and choice

S.NO	Type of Question	To be given in the Question Paper			To be answered		
		No. of Questions	Marks allotted to each question	Total Marks	No. of Questions	Marks allotted to each question	Total Marks
1	Section-I Part-A and Part-B Essay Questions	6	10	60	3	10	30
2	Section-II Short Questions	7	5	35	4	5	20
TOTAL		13		95	TOTAL MARKS		50

$$\text{Percentage of choice given} = \frac{95 - 50}{95} \times 100 = \frac{45}{95} \times 100 = 47.36\%$$

Model Blue print for the question paper setter

Chapter Name	Essay Questions 10 Marks	Short Questions 5 Marks	Marks allotted to the chapter
MODULE -I	2	2	30
MODULE -II	1	2	20
MODULE -III	1	1	15
MODULE -IV	1	1	15
MODULE -V	1	1	15
Total No. of questions	6	7	
Total Marks Including choice			95

PR GOVT COLLEGE (A): KAKINADA
DEPARTMENT OF COMPUTER APPLICATIONS
II B.Sc (IT)-SEMESTER-IV
OPERATING SYSTEM

Time: 2 Hrs

Max. Marks: 50

SECTION - I

Answer any three of the following questions. Must attempt atleast one question from each part. Each question carries 10 Marks. 3 X 10 = 30M

Part-A

1. What is Operating System? Explain functions of Operating System.
2. Explain about Process Scheduling Algorithms in detail?
3. Explain about Deadlock Detection and recovery?

Part-B

4. Explain the various types of Operating System?
5. Explain about classical Process Synchronization problems?
6. Explain about Segmentation and Memory Allocation Strategies?

SECTION - II

Answer any four of the following questions. Each question carries 5 marks. 4 X 5 = 20M

7. Explain Real Time System and Time sharing Operating System?
8. Explain about Thread Concept in detail?
9. Write about the Process and the Process state?
10. Explain Threading issues?
11. Write about some necessary and sufficient conditions for Deadlock?
12. Explain about Virtual memory?
13. Write about file Operations?

PR GOVERNMENT COLLEGE (AUTONOMOUS), KAKINADA					
Course Code 25IT42	COMPUTER NETWORKS	II BSC(IT) Sem – IV Major-9 2026-27			
Hours	75	L	T	P	C
Pre requisites	Networking Basics	3	-	1	4

Course Objectives

1. Introduce foundational concepts and architecture of computer networks, including OSI and TCP/IP models.
2. Explain the functionalities of network layers, from physical to application, highlighting their roles in communication systems.
3. Explore data transmission technologies, including guided and wireless media, and assess their real-world applications.
4. Develop understanding of core protocols, error handling mechanisms, and routing algorithms used in modern networks.
5. Demonstrate how network services operate, focusing on protocols like TCP, UDP, HTTP, DNS, and their relevance in global connectivity.

Course Outcomes

These define what students should be able to do after successful completion, At the End of the Course, The Students will be able to:

Outcome	Description	Cognitive Level
C01	Describe network models (OSI, TCP/IP) and differentiate between network hardware and software components.	Understand
C02	Analyze data transmission techniques and select appropriate media for specific networking scenarios.	Analyze
C03	Apply error control and flow protocols (e.g., sliding window, ALOHA) to optimize link-layer communication.	Apply
C04	Evaluate routing strategies and congestion control algorithms within network environments, including Internet-based systems.	Evaluate
C05	Implement basic application-layer protocols and illustrate how services like email, web browsing, and streaming are supported on networks.	Apply

P R GOVT COLLEGE(AUTONOMOUS), KAKINADA
DEPARTMENT OF COMPUTER APPLICATIONS
II BSC(IT) Semester-IV (W.E.F. 2026-27)
COMPUTER NETWORKS

SYLLABUS

Theory

Credits: 3

3 hrs/week

Unit 1. Introduction to Computer Networks:

Network hardware, Network software, OSI, TCP/IP Reference models, Example Networks: ARPANET, Internet.

Physical Layer: Guided Transmission media: twisted pairs, coaxial cable, fiber optics, Wireless transmission.

Unit 2. Data link layer:

Design issues, framing, Error detection and correction. Elementary data link protocols, Sliding Window protocols.

Medium Access sub layer: The channel allocation problem.

Multiple access protocols: ALOHA, Carrier sense multiple access protocols, collision free protocols.

Wireless LANs, Data link layer switching.

Unit 3. Network Layer:

Design issues, Routing algorithms: shortest path routing, Flooding, Hierarchical routing, Broadcast, Multicast, distance vector routing, Congestion Control Algorithms, Quality of Service, Internetworking, The Network layer in the internet.

Unit 4. Transport Layer:

Transport Services, Elements of Transport protocols, Connection management, TCP and UDP protocols.

Unit 5. Application Layer:

Domain name system, SNMP, Electronic Mail, SMTP, World Wide Web, HTTP and HTTPS.

Textbooks:

1. Computer Networks -- Andrew S Tanenbaum, David. j. Wetherall, 5th Edition. Pearson Education/PHI
2. Data Communications and Networking – Behrouz A. Forouzan. Third Edition TMH.

Reference Books:

1. An Introduction to Computer Networks- Peter Lars Dordal, Loyola University Chicago (2022); eBook (Creative Commons Licensed)
2. The TCP/IP Guide: A comprehensive, Illustrated Internet Protocols reference, Charles M. Kozierok

P R GOVT COLLEGE(AUTONOMOUS), KAKINADA
DEPARTMENT OF COMPUTER APPLICATIONS
II BSC(IT) Semester-IV (W.E.F. 2026-27)
COMPUTER NETWORKS

Practical

Credits: 1

2 hrs/week

List of Experiments:

1. Understanding various network tools in Windows and Linux
2. Study different types of Network devices and Cables
3. Building a Local Area Network.
4. Concept of Network IP Address
5. Introduction to Network Simulator – Packet Tracer (PT)
6. Configuration of a Router using Packet Tracer
7. Implementation of a Network using Packet Tracer
8. Implementation of Static Routing using Packet Tracer
9. Implementation of RIP using Packet Tracer
10. Implementation of OSPF using Packet Tracer
11. Implement DNS using packet tracer
12. Implementation of a VLAN using Packet Tracer

P R GOVT COLLEGE(AUTONOMOUS), KAKINADA
DEPARTMENT OF COMPUTER APPLICATIONS
II BSC(IT) Semester- IV (2026-2027)
COMPUTER NETWORKS

PAPER- 9

Marks 50M

Model blue print for the model paper and choice

S.NO	Type of Question	To be given in the Question Paper			To be answered		
		No. of Questions	Marks allotted to each question	Total Marks	No. of Questions	Marks allotted to each question	Total Marks
1	Section-A Essay Questions	6	10	60	3	10	30
2	Section-B Short Questions	7	5	35	4	5	20
TOTAL		13		95	TOTAL MARKS		50

$$\text{Percentage of choice given} = \frac{95 - 50}{95} \times 100 = \frac{45}{95} \times 100 = 47.36\%$$

Model Blue print for the question paper setter

Chapter Name	Essay Questions 10 Marks	Short Questions 5 Marks	Marks allotted to the chapter
UNIT-I	2	2	30
UNIT -II	1	2	20
UNIT -III	1	1	15
UNIT -IV	1	1	15
UNIT -V	1	1	15
Total No. of questions	6	7	
Total Marks Including choice			95

P R GOVT COLLEGE(AUTONOMOUS), KAKINADA

DEPARTMENT OF COMPUTER APPLICATIONS

II BSC(IT) Semester-IV (W.E.F. 2026-27)

MODEL PAPER

COMPUTER NETWORKS

Time: 2 Hrs

SEMESTER - IV

Max Marks: 50

SECTION-I

Answer Any Three Questions. At least One question from each part (3x10=30)

Part-A

1. Explain the OSI Reference Model with a neat diagram and describe the functions of each layer.
2. Explain the TCP/IP Reference Model and compare it with the OSI Reference Model.
3. Explain the Sliding Window Protocol with a neat diagram and its working.

Part-B

4. Explain the Shortest Path Routing Algorithm and Distance Vector Routing Algorithm with suitable examples.
5. Explain the TCP and UDP Protocols. Compare their features and applications.
6. Explain the Domain Name System (DNS) and describe the working of HTTP and HTTPS.

SECTION-II

Answer any FOUR Questions. Each question carries 5 marks (4x5=20)

7. Explain the Network Hardware and its types.
8. Write short notes on Transmission Media (Twisted Pair, Coaxial Cable, Fiber Optic and Wireless Transmission).
9. What is Framing? Explain its types.
10. Explain the ALOHA Protocol.
11. What is Flooding Routing?
12. What is Connection Management in the Transport Layer?
13. Differentiate between HTTP and HTTPS.

SEMESTER-5

PR GOVERNMENT COLLEGE (AUTONOMOUS), KAKINADA							
Course Code 24IT51	R Programming			III B.Sc- (IT) Major 12 Sem - V 2026-27			
Hours	90 (60 Theory + 30 Practical)			L	T	P	C
Pre requisites	Basic Knowledge of Computer			3	-	2	4

Course Objective
The objective of this course is to provide students 1.To understand installation of R, variable names and operators in R 2.To understand control structures, data structures of R 3.To understand vectors and their operations 4.To understand Matrices, Lists, Data frames etc

Course Outcomes	
Students after successful completion of the course will be able to:	
CO1	1. Gain a solid understanding of R programming language
CO2	2. Acquire knowledge on various data structures and control structures in R.
CO3	3Perform vectorized operations in R programming.
CO4	4. Develop skills in manipulating and transforming vectors, matrices, arrays and lists in R.
CO5	5. Explore and analyze data using data frames and tables.

P R GOVT COLLEGE(AUTONOMOUS), KAKINADA
DEPARTMENT OF COMPUTER APPLICATIONS
III B.Sc (IT) Semester- V(2026-27), Major 12
R-PROGRAMMING
SYLLABUS

UNIT I

Introduction to R: R overview and history, Basic features of R, Benefits of R, data types in R, Installing R, Getting started with the RStudio IDE, Running R, Packages in R, variable names and assignment, operators, Input/output functions, reading and writing data.

UNIT II

Control structures: Conditional statements, Loops, dates and times, functions, String manipulations. Preview of Some Important R Data Structures: Vectors, Character Strings, Matrices, Lists, Data Frames, and Classes.

UNIT III

Vectors: Scalars, Vectors, Arrays and Matrices: Adding and Deleting Vector Elements, Obtaining the Length of a Vector Common vector operations: Arithmetic & logical operations, Vector Indexing, Generating vector sequences with seq(), Repeating vector constants with rep (), using all () and any () functions, Vectorized operations, NA and NULL values.

UNIT IV

Matrices and Arrays: Creating Matrices, General Matrix operations-linear algebra operations, matrix indexing, filtering on matrices, using apply () function , Add and Delete matrix rows and columns. Lists: Creating Lists, General List Operations, List Indexing Adding and Deleting List Elements, Getting the Size of a List ,Accessing List Components and Values, Using l apply()and apply()functions.

UNIT-V

Data Frames: Creating Data Frames, Accessing Data Frames-Other Matrix-Like Operations: Extracting sub data frames, using r bind()and c bind()functions.

Factors and Tables : Factors and Levels - Common Functions Used with Factors : t apply() ,split() and by() - Working with Tables, Matrix/Array-Like Operations on Tables, Extracting a Sub table-Math Functions: aggregate()and cut()functions.

Additional inputs: Data visualization in R

III. REFERENCES :

TEXTBOOKS:

1. The Art of R Programming by Norman Matloff, Nostarch press, sanfransisco,2011.
2. An Introduction to R for Beginners by SASHA HAFNER,on AUG-2019

REFERENCE BOOKS:

1. R Programming for Dummies, Andrie de Vries and Joris Meys, Wiley
2. R for Data Science, Hadley Wickham, Garrett Grolemund ,O'Reilly Media
3. R Programming: A Step-By-Step Guide for Absolute Beginners-2nd Edition, Daniel Daniel Bell
4. Learn R programming in 1 Day, Krishna Rungta, Published by Guru99

IV.SUGGESTED CO-CURRICULAR ACTIVITIES:

1. Assign students real-world data analysis projects that require them to apply their programming skills.
2. Organize coding challenges focused on R Programming.

3. Organize guest lectures or workshops.
4. William Stalling: Computer Organization and Architecture. Prentice Hall
5. Rajaraman& T. Radhakrishnan: Computer Organization and Architecture. PHI
6. Donald D. Givone: Digital Principles and Design. McGraw Hill

V Semester
Course 12: R Programming Practical

- 1) Write an R Program to take in put from user.
- 2) Write an R Program to demonstrate working with operators (Arithmetic, Relational, Logical, Assignment operators).
- 3) Write an R Program to Check if a Number is Odd or Even
- 4) Write an R Program to check if the given Number is a Prime Number
- 5) Write an R Program to Find the Factorial of a Number
- 6) Write an R Program to Find the Fibonacci sequence Using Recursive Function
- 7) Write an R Program to create a Vector and to access elements in a Vector
- 8) Write an R Program to create a Matrix from a Vector using dim()function.
- 9) Write an R Program to create a List and modify its components.
- 10) Write an R Program to create a Data Frame.
- 11) Write an R Program to access a Data Frame like a List.
- 12) Write an R Program to create a Factor

P R GOVT COLLEGE(AUTONOMOUS), KAKINADA
DEPARTMENT OF COMPUTER APPLICATIONS
III B.Sc (IT) Semester- V (2026-27)
R-PROGRAMMING

Marks: 50M

Model blue print for the model paper and choice

S.NO	Type of Question	To be given in the Question Paper			To be answered		
		No. of Questions	Marks allotted to each question	Total Marks	No. of Questions	Marks allotted to each question	Total Marks
1	Section-A Essay Questions	6	10	60	3	10	30
2	Section-B Short Questions	7	5	35	4	5	20
TOTAL		13		95	TOTAL MARKS		50

Percentage of choice given = $\frac{95 - 50}{95} \times 100 = \frac{45}{95} \times 100 = 47.36\%$

Model Blue print for the question paper setter

Chapter Name	Essay Questions 10 Marks	Short Questions 5 Marks	Marks allotted to the chapter
UNIT-I	2	2	30
UNIT -II	1	2	20
UNIT -III	1	1	15
UNIT -IV	1	1	15
UNIT -V	1	1	15
Total No. of questions	6	7	
Total Marks Including choice			95

P R GOVT COLLEGE(AUTONOMOUS), KAKINADA
DEPARTMENT OF COMPUTER APPLICATIONS
III B.Sc (IT) Semester- V (2026-27)

R-PROGRAMMING

MODEL PAPER

Time : 2Hrs

SEMESTER-V

Max. Marks: 50

SECTION-I

Answer Any Three Questions. At least One question from each part (3x10=30M)

Part-A

1. How do you install a package in R programming?
2. Explain about Looping statements in R Programming with examples
3. What is a Vector? Explain about vector operations in R programming with an examples?

Part-B

4. Explain the basic features of R Programming Language
5. Explain about list operations with an examples.
6. What is a Data Frame? Explain the functions of a Data Frame.

SECTION-II

Answer any FOUR Questions. Each question carries 5 marks (4x5=20M)

7. What are the Data types available in R language?
8. What are the advantages of R-programming language?
9. Write about next and break in R programming?
10. Write about character strings in R programming?
11. Explain about adding and deleting vector elements. With an examples
12. How to create matrix in R programming?
13. Define the factors. give an example?

P.R. GOVT. COLLEGE (A), KAKINADA
III B.Sc (IT) (W.E.F 2026-27)
SEMESTER – V
QUESTION BANK

Subject: R – Programming

Paper: 12

MODULE-I

Essay Questions:

1. Explain the basic features of R Programming Language
2. Explain the installation procedure of R and R Studio IDE?
3. Explain Different types of Operators ((arithmetic, logical, relational and , assignment operators) in R Language with examples?
4. How do you install a package in R programming?

Short Questions:

5. What are the Data types available in R language?
6. Explain Relational operators in R language.
7. Write about input/output functions in R?
8. How do you assign a variable in R-programming?
9. What are the advantages of R-programming language?

MODULE-II

Essay Questions:

1. Explain about Looping statements in R Programming with examples
2. What are the Conditional Statements available in R Programming? Explain with examples?
3. Write about string manipulation functions that are available in R Programming.

Short Questions:

4. Write a while control structure in a R programming with an example?
5. Write about next and break in R programming?
6. How to declare the date and time functions in R programming?
7. Write about character strings in R programming?

MODULE-III

Essay Questions:

1. What is a Vector? Explain about vector operations in R programming with an examples.
2. Explain about any five functions related to vectors in R programming
3. Explain about arithmetic and logical operations on vectors?

Short Questions:

4. How to find the length of a vector with an example?
5. How to extracting elements of a vector using subscripts?
6. Explain about adding and deleting vector elements. With an examples
7. Explain about generating sequences in R programming

MODULE-IV

Essay Questions:

1. Write about matrix operations in R programming?
2. Explain about list operations with an examples.
3. Explain about creating and adding of elements to the list give an example?

Short Questions:

1. How to create the list in R programming?.
2. How to create matrix in R programming?
3. Explain about accessing and deleting elements from the list give an example?
8. How to find the size of a list with an example?

MODULE-V

Essay Questions:

1. What is a Data Frame? Explain the functions of a Data Frame.
2. Explain about factor operations in R?
3. Explain about mathematical functions in R..

Short Questions:

4. How to create data frames in a R programming?
5. How to access the elements from the data frame in R programming?
6. Define the factors. give an example?
7. Write about generating factor levels in R programming

Mapping of Cos with POs/PSOs

[illegible]

PR GOVERNMENT COLLEGE (AUTONOMOUS), KAKINADA							
Course Code 24IT52	SOFTWARE ENGINEERING			III - BSC(IT) (MAJOR-13) SEM - V 2026-27			
Hours	90 (60 Theory + 30 Practical)			L	T	P	C
Pre requisites	BASIC COMPUTER KNOWLEDGE			3	-	1	4

Course Objective

To introduce Object-oriented software engineering (OOSE) - which is a popular technical approach to analyzing, designing an application, system, or business by applying the object-oriented paradigm and visual modeling

Course Outcomes:

On Completion of the course, the students will be able to-		Cognitive Domain
C01	Understand and apply the fundamental principles of Object-Oriented Programming (OOP) concepts and Unified Modeling Language (UML) basics, in the development of software solutions.	Understanding
C02	Analyze and specify software requirements, develop use cases and scenarios, apply object- oriented analysis and design (OOAD) principles	Application
C03	Familiar with the concept of test-driven development(TDD) and its practical implementation	Application
C04	Analyze and Evaluate Software Maintenance and Evolution Strategies	Application
C05	Apply Advanced Object-Oriented Software Engineering Concepts	Application

PR GOVERNMENT COLLEGE(A), KAKINADA
DEPARTMENT OF COMPUTER APPLICATIONS
III BSC(IT) (MAJOR) SEMISTER-V (W.E.F.2026-2027)
COURSE 13: SOFTWARE ENGINEERING

UNIT-I

Introduction to Object-Oriented Programming: Overview of software engineering, Introduction to Object-Oriented Programming (OOP) concepts (classes, objects, inheritance, polymorphism), Unified Modelling Language (UML) basics, Introduction to software development process and software development life cycle (SDLC).

UNIT-II

Requirements Analysis and Design: Requirements analysis and specification, Use cases and scenarios, Object-oriented analysis and design (OOAD), Design patterns, UML modelling techniques (class diagrams, sequence diagrams, state machine diagrams, activity diagrams)

UNIT-III

Software Construction and Testing: Software construction basics, Object-oriented design principles, Object-oriented programming languages (Java, C++, Python), Software testing basics (unit testing, integration testing, system testing), Test-driven development (TDD)

UNIT-IV

Software Maintenance and Evolution: Software maintenance basics, refactoring techniques Software version control, Code review and inspection, Software evolution and reengineering

UNIT-V

Advanced Topics in Object-Oriented Software Engineering: Model-driven engineering (MDE), Aspect-oriented programming (AOP), Component-based software engineering (CBSE), Service-oriented architecture (SOA), Agile software development and Scrum Methodologies

Text Book(s)

1. An Introduction to Object-Oriented Analysis and Design and the Unified Process, 3rd Edition, Craig Larman, Prentice-Hall.
2. Programming in Java by Sachin Malhotra, Oxford University Press

Reference Books

1. Requirements engineering: processes and techniques, G. Kotonya and, I. Sommerville, 1998, Wiley
2. Design Patterns, E. Gamma, R. Helm, R. Johnson, and J. Vlissides
3. The Unified Modeling Language Reference Manual, J. Rumbaugh, I. Jacobson and G. Booch, Addison Wesley

SUGGESTED CO-CURRICULAR ACTIVITIES:

1. Assign students real-world software development projects that require them to apply software engineering principles and practices.
2. Encourage students to prepare and deliver technical presentations or demonstrations on software engineering topics of their choice.
3. Invite industry professionals and experienced software engineers to deliver guest lectures or conduct workshops

COURSE 13: SOFTWARE ENGINEERING

Practical

Credits: 1

2 hrs/week

1. To perform the Requirement analysis of the specified problem and draw a flowchart.
2. Understanding of System modeling: Data model i.e. ER – Diagram and draw The ER Diagram with generalization, specialization and aggregation of specified problem statement.
3. Understanding of System modeling: Functional modeling: DFDlevel0i.e. Context Diagram and draw it.
4. Understanding of System modeling: Functional modeling: DFDlevel1and DFDlevel2and draw it.
5. Understanding different actors and use cases in detail of the specified problem statement and draw it. Using Rational Rose software any other available software.
6. To perform the user's view analysis: Use case diagram and draw it using Rational Rose or any other available software.
7. To draw the structural view diagram: Class diagram of specified problem statement using Rational Rose or any other available software.
8. To draw the behavioral view diagram: State-chart diagram, Activity diagram of specified problem Statement using Rational Rose any other available software.
9. To understand testing and perform Boundary value analysis and Equivalence class testing.
10. To draw Flow graph, DD paths , calculation of cyclomatic complexity and find out all the Independent paths from the D D paths graph.
11. Case study: Prepares RS for a given problem statement.

P. R.GOV.T. COLLEGE (AUTONOMOUS), KAKINADA
MODEL BLUE PRINT FOR THE YEAR 2026-2027
B.SC (IT) 2026-2027 BATCH
SEMESTER-V

SUBJECT: SOFTWARE ENGINEERING
Time: 2 Hrs.

PAPER: 13
Max.Marks: 50

Model blue print for the model paper and choice

S.NO	Type of Question	To be given in the Question Paper			To be answered		
		No. of Questions	Marks allotted to each question	Total Marks	No. of Questions	Marks allotted to each question	Total Marks
1	Section-I Part-A and Part-B Essay Questions	6	10	60	3	10	30
2	Section-II Short Questions	7	5	35	4	5	20
TOTAL		13		95	TOTAL MARKS		50

$$\text{Percentage of choice given} = \frac{95 - 50}{95} \times 100 = \frac{45}{95} \times 100 = 47.36\%$$

Blue Print

Chapter Name	Essay Questions 10 Marks	Short Questions 10 Marks	Marks allotted to the chapter
UNIT -I	2	2	30
UNIT -II	1	1	15
UNIT -III	1	1	15
UNIT -IV	1	2	20
UNIT -V	1	1	15
Total No. of questions	7	6	
Total Marks Including choice			95

P.R.GOV.T.COLLEGE (AUTONOMOUS), KAKINADA
DEPARTMENT OF COMPUTER APPLICATIONS
MODEL PAPER
III B.Sc(IT)

SUBJECT: SOFTWARE ENGINEERING
Time: 2 Hrs

PAPER: 13
Marks: 50

SEMESTER-V

SECTION –I

Answer any three of the following questions. Must attempt atleast one question from each part.

Each question carries 10 Marks.

3 X10 =30

Part-A

1. Define Software Engineering? Discuss about features of OOP in OOSE.[BT1,2]
2. Explain about the Software Requirements Analysis and explain its characteristics[.BT1,2]
3. Define Software Testing? Explain the various types of Software Testing's ?[BT1,3]

Part-B

4. Explain the about Software Development Life Cycle (SDLC) phases? [BT1]
5. Demonstrate on the various activities in Software Maintenance.[BT1]
6. Define MDE? Explain the various characteristics of MDE.[BT1,2]

SECTION – II

Answer any four of the following questions. Each question carries 5 m. 4 X 5 = 20M

7. Discuss about Structural Modeling and Behavioral Modeling diagram in UML?[BT1,2]
8. Define Design pattern? List out various types of Design pattern[BT1,2]
9. Demonstrate on the OOA ?[BT1,2]
10. Explain about Test-Driven Development (TDD)? [BT1]
11. Write about Software Construction fundamentals? [BT1,2]
12. Define Code Inspection? And Explain it.[BT1,2]
13. Explain about various phases of Agile Software Development?[BT2]

P R GOVERNMENT (AUTONOMOUS) COLLEGE, KAKINADA
QUESTION BANK (W.E.F. 2026-2027)
COMPUTER APPLICATIONS
SOFTWARE ENGINEERING
SEMESTER-V

Module I

Essay Questions: (10M)

1. What is Object Oriented Programming? Explain the important Principles? [BT1,2]
2. Explain the about Software Development Life Cycle (SDLC) phases ? [BT1]
3. Define UML? Categorize various types of UML in OOSE. [BT1,2]

Short Questions: (5M)

4. Explain the terms Inheritance & Polymorphism in OOP? [BT1,2]
5. Demonstrate on Software development Process? [BT1,2]
6. What are the various diagrams used in UML? [BT1,2]

Module II

Essay Questions: (10M)

1. Explain about the Software Requirements Analysis? [BT1,2]
2. Demonstrate on the OOA and OOD? [BT1,2]
3. Demonstrate on the various UML modeling techniques ?

Short Questions: (5M)

4. Define Design pattern? List out various types of Design pattern [BT1,2]
5. Discuss about Use case diagram? and Explain giving an example? [BT1]
6. Explain about UML Activity diagram & What are the symbols used in an activity diagram?

Module III

Essay Questions: (10M)

1. Demonstrate on the Object Oriented Programming Languages in OOSE? [BT1,2]
2. Define Software Testing? Explain the various types of Software Testing's ? [BT1,3]

Short Questions: (5M)

3. Write about Software Construction fundamentals? [BT1,2]
4. Write about any two Software Testing's? [BT1,2]
5. Explain about Test-Driven Development (TDD)? [BT1]

Module IV

Essay Questions: (10M)

1. Demonstrate on the various activities in Software Maintenance.[BT1]
2. Define Refactoring? What are the techniques used for it [BT1,2]
3. Define Software Version? What is software version control? Explain [BT1,2]

Short Questions: (5M)

4. Define Code Inspection? And Explain it.[BT1,2]
5. Define Software Evaluation?And Explain it .[BT1,2]
6. What are Software reengineering activities?[BT1,2]

Module V

Essay Questions: (10M)

1. Define MDE? Explain the various characteristics of MDE.[BT1,2]
2. Define CBSE ?Explain its framework activities ?[BT1,2]
3. Define Scrum ?Demonstrate on Scrum methodologies and Principles ?[BT1,2]

Short Questions: (5M)

4. Write about Service Oriented Architecture(SOA) [BT1,2]
5. Demonstrate on Aspect Oriented Programming(AOP) and its importance?[BT2]
6. Explain about various phases of Agile Software Development?[BT2]

PR GOVERNMENT COLLEGE (AUTONOMOUS), KAKINADA						
Course Code 24IT53	Computer Networks			III B.Sc. (IT) Sem-V Major 14 2026-27		
Hours	90 (60+ 30)			L	T	P C
Pre requisites	Basic Computer Knowledge			3	-	1 4

Course Objective
1. Understand the basics of Data Communication 2. Understand the OSI layer. 3. Understand different protocols.

Course Outcomes

On completion of the course, the students will be able to-		
Outcome	Description	Cognitive Level
CO1	Understand and apply network applications, hardware, software, and reference models for network communication.	Knowledge
CO2	Design and analyze data link layer protocols, multiple access protocols, and wireless LAN technologies.	Knowledge
CO3	Design routing algorithms, congestion control algorithms, and evaluate network layer protocols for internetworking.	Analysis and Evaluation
CO4	Analyze transport service, transport protocols, and evaluate UDP and TCP in the internet.	Application
CO5	Understand and evaluate application layer protocols, including DNS, email, WWW, and network Management protocols.	Creativity

P.R. GOVT. COLLEGE (AUTONOMOUS), KAKINADA
DEPARTMENT OF COMPUTER APPLICATIONS
III B.Sc.–IT Semester–V(W.E.F.2026-27)

Computer Networks
SYLLABUS

UNIT-I

INTRODUCTION: Network applications, network hardware, network software, reference models : OSI, TCP/IP, Internet, Connection oriented network X.25, frame relay.

THE PHYSICAL LAYER: Theoretical basis for communication, guided transmission media, wireless transmission, the public switched telephone networks, mobile telephone system.

UNIT-II

DATA LINK LAYER: Design issues, error detection and correction, elementary data link protocols, sliding window protocols, example datalink protocols -HDLC, the datalink layer on the internet.

MEDIUM ACCESS SUB LAYER: Channel allocations problem, multiple access protocols, Ethernet, Data Link Layer switching, Wireless LAN, Broadband Wireless, Bluetooth.

UNIT-III

NETWORK LAYER: Network layer design issues, routing algorithms, Congestion control algorithms, Internet working, the network layer in the internet (IPv4andIPv6), Quality of Service.

UNIT-IV

TRANSPORT LAYER: Transport service, elements of transport protocol, Simple Transport Protocol, Internet transport layer protocols: UDP and TCP.

UNIT-V

APPLICATION LAYER: Domain name system, electronic mail, World Wide Web: architectural overview, dynamic web document and http.

APPLICATION LAYER PROTOCOLS: Simple Network Management Protocol, File Transfer Protocol, Simple Mail Transfer Protocol, Telnet.

REFERENCES TEXT BOOKS

1. A. S. Tanenbaum (2003), Computer Networks, 4th edition, Pearson Education / PHI, New Delhi, India

REFERENCE BOOKS

1. Behrouz A. Forouzan (2006), Data communication and Networking, 4th Edition, McGraw Hill, India.
2. Kurose, Ross (2010), Computer Networking: A top-down approach, Pearson Education, India.

SUGGESTED CO-CURRICULAR ACTIVITIES:

1. Hands-on exercises to configure network applications
2. Guest Lectures and Workshops on routing algorithms, congestion control, and network layer protocols
3. Group Projects on Network Application Development

P.R. GOVT. COLLEGE (AUTONOMOUS), KAKINADA

DEPARTMENT OF COMPUTER APPLICATIONS

III B.Sc. – IT Semester – V (W.E.F. 2026-27)

Computer Networks
PRACTICAL

1. Introduction to networking tools and Linux
2. Introduction to Packet Tracer tool from Cisco
3. Study different types of network cables
4. Study different types of networks in detail
5. Study the basics of TCP/IP using various networking tools available in Linux
6. Create a network topology using packet tracer
7. Configure routing using packet tracer
8. Study network security algorithms
9. Implements DNS using packet tracer
10. Implement SMTP connectivity

P.R.GOV.T. COLLEGE (AUTONOMOUS), KAKINADA
DEPARTMENT OF COMPUTER APPLICATIONS
(W.E.F.2026-27) III B.Sc- (IT) SEMESTER-V

SUBJECT: Computer Networks

Max.Marks:50

Model blue print for the model paper and choice

S.NO	Type of Question	To be given in the Question Paper			To be answered		
		No. of Questions	Marks allotted to each question	Total Marks	No. of Questions	Marks allotted to each question	Total Marks
1	Section-I Essay Questions	6	10	60	3	10	30
2	Section-II Short Questions	7	5	35	4	5	20
TOTAL		13		95	TOTAL MARKS		50

$$\text{Percentage of choice given} = \frac{95 - 50}{95} \times 100 = \frac{45}{95} \times 100 = 47.36\%$$

Model Blue print for the question paper Setter

Chapter Name	Essay Questions 10 Marks	Short Questions 5 Marks	Marks allotted to the chapter
UNIT-I	2	2	30
UNIT-II	1	2	20
UNIT-III	1	1	15
UNIT-IV	1	1	15
UNIT-V	1	1	15
Total No. of questions	6	7	
Total Marks Including choice			95

P.R. GOVT. COLLEGE (AUTONOMOUS), KAKINADA

DEPARTMENT OF COMPUTER APPLICATIONS

MODEL PAPER (W.E.F.2026-27)

III B.Sc - IT Semester – V

COMPUTER NETWORKS

Time:2 Hrs.

SEMESTER- V

Max.Marks:50

SECTION-I

Answer Any Three Questions. At least One question from each part

(3x10=30)

Part-A

1. Explain the OSI reference model in detail
2. Explain about Network Devices with examples
3. Explain about Datalink layer and its design issues

Part-B

4. Discuss about Congestion Control techniques in Network layer
5. Write about Transport layer and its services
6. Briefly discuss about FTP & SMTP

SECTION-II

Answer any FOUR Questions. Each question carries 5 marks

(4x5=20)

7. Write about Guided media with example
8. Explain about X.25 frame relay
9. What is Sliding window Protocol
10. Explain about Ethernet
11. What is IPv4 & IPV6
12. Write a short note on Simple transport Protocol (STP)
13. What is TELNET

Question Bank

Essay Questions

1.	Explain the OSI reference model in detail	BT2	CO1	PO1
2.	Explain the TCP/IP reference model in detail	BT3	CO4	PO2
3.	Explain about Network Devices with examples	BT1	CO4	PO3

1.	Explain about Datalink layer and its design issues	BT2	CO2	PO5
2.	Write about HDLC protocol in detail	BT2	CO1	PO1
3.	Write about error detection and correction in datalink layer?	BT3	CO4	PO2

1.	Explain about design issues of Network layer	BT3	CO4	PO1
2.	Discuss about Congestion Control techniques in Network layer	BT2	CO4	PO2
3.	Describe the role of (IPV4 & IPV6)? Highlight the differences?	BT1	CO2	PO1

1.	Write about Transport layer and its services	BT1	CO1	PO1
2.	Differentiate UDP & TCP in Transport layer	BT2	CO1	PO1

1.	Discus about DNS? Explain its working in networking	BT2	CO3	PO2
2.	Briefly discuss about FTP & SMTP	BT4	CO2	PO4

Shorts Questions

1.	Explain about X.25 frame relay	BT2	CO4	PO2
2.	Write about Guided media with example	BT1	CO2	PO1
3.	Write about Wireless transmission media	BT2	CO1	PO1

PR GOVERNMENT COLLEGE(AUTONOMOUS), KAKINADA					
Course Code: 24IT54	Mobile Application Development	III B.Sc.(IT) Major 15 Sem -V 2026-27			
Hours	75	L	T	P	C
Pre requisites	Basic Computer Knowledge	3	-	1	4

Course Objective
1. Understand the importance of Mobile Application Development 2. Understand the Android studio and Flutter. 3. Apply programming skills to develop Mobile applications.

Course Outcomes

On completion of the course, the students will be able to		
Outcome	Description	Cognitive Level
CO1	Learn-to setup an ew Material Appusing Android Studio.	Knowledge
CO2	Understand the Widget tree and learn trouser-made Flutter, widgets for user interface design.	Knowledge
CO3	Learn to incorporate Image and Text Widgets to create simple user interfaces.	Analysis and Evaluation
CO4	Learn to customize pre-built Flutter widgets.	Application
CO5	Adding App Icons for iOS and Android builds.	Creativity

P.R.GOV.T.COLLEGE(AUTONOMOUS),KAKINADA
DEPARTMENT OF COMPUTER APPLICATIONS

III B.Sc.–IT Semester–V(W.E.F.2026-27)
Mobile Application Development
SYLLABUS

UNIT:I

Introduction: Frame works and Tools for Mobile App Development, Characteristics of Mobile Applications, History of Mobile Application Frameworks and Tools, Introduction to Android, iOS, and Flutter. Client-Server Architecture: 1-tier, 2-tier, 3-tier, types of Connection, Synchronization, Mobile Device Types, Mobile Device Components, Types of Mobile Applications.

UNIT:II

Mobile Application Development using Flutter: to set up a new Material App using Android Studio, Creating UI with Flutter: Using Hot Reload and Hot Restart to quickly refresh the app UI and understand when to use each, using the Pub spec. yaml file to incorporate dependencies, custom assets and fonts, an introduction to the Widget build() method, using layout widgets such as Columns, Rows, Containers and Cards, incorporating Material icons using the Icons class.

UNIT:III

Building Apps with State: Understanding the difference between Stateful and Stateless widgets and when they should each be used, understanding how callbacks can be used to detect user interaction in button widgets, declarative style of UI programming and how Flutter widgets react to state changes, importing dart libraries to incorporate additional functionality, variables, data types and functions work in Dart, building flexible layouts using the Flutter Expanded widget, relationship between setState(), State objects and Stateful Widgets.

UNIT:IV

Using the Dart package manager: to use Dart package manager to incorporate Flutter compatible packages into your projects, functions in Dart and the arrow syntax, to refactor widgets and understand Flutter's philosophy of UI as code. Structuring Flutter Apps: to use Dart Constructors to create customisable Flutter widgets, apply common mobile design patterns to structure Flutter apps.

UNIT:V

Security: User to Mobile Client Security Issues, Mobile Client Security Issues, Client-Server Communications Security Issues, Existing Web Architectures and Back-End Systems Security Issues, Mobile Application Development Management

REFERENCES

1. Lee, H. Schneider, and R. Schell: Mobile Applications: Architecture, Design, and Development. Pearson
2. Marco L. Napoli: Beginning Flutter: A Hands-on Guide to App Development. Wiley
3. Bill Phillips & Brian Hardy: Android Programming the Big Nerd Ranch Guide Big Nerd Ranch
4. Brian Fling : Mobile Design and Development. O'Reilly

P.R.GOV.T.COLLEGE(AUTONOMOUS), KAKINADA

DEPARTMENT OF COMPUTER APPLICATIONS

III B.Sc. – IT Semester – V (W.E.F. 2026-27)

Mobile Application Development

PRACTICAL

1. Develop a program to implement frame layout, table layout and relative layout.
2. Develop a program to implement Text View and Edit Text.
3. Develop a program to implement Auto Complete Text View.
4. Develop a program to implement Button. Image Button and Toggle Button.
5. Develop a program to implement login window using the above UI controls.
6. Develop a program to implement Checkbox
7. Develop a program to implement Radio Button and Radio Group.
8. Develop a program to implement Progress Bar.
9. Develop a program to implement List View, Grid View, Image View and Scroll View.
10. Develop a program to implement Date and Time Picker

.
Mobile application Project Ideas

i) Tic Tac Toe Game

P.R.GOV.T.COLLEGE (AUTONOMOUS),KAKINADA

DEPARTMENT OF COMPUTER APPLICATIONS

MODEL BLUE PRINT(W.E.F.2026-27)

III B.Sc- (IT) SEMESTER-V

Model blue print for the model paper and choice

S.NO	Type of Question	To be given in the Question Paper			To be answered		
		No. of Questions	Marks allotted to each question	Total Marks	No. of Questions	Marks allotted to each question	Total Marks
1	Section-I Essay Questions	6	10	60	3	10	30
2	Section-II Short Questions	7	5	35	4	5	20
TOTAL		13		95	TOTALMARKS		50

Model Blue print for the question paper Setter

Chapter Name	Essay Questions 10 Marks	Short Questions 5 Marks	Marks allotted to the chapter
UNIT-I	2	2	30
UNIT-II	1	2	20
UNIT-III	1	1	15
UNIT-IV	1	1	15
UNIT-V	1	1	15
TotalNo.of questions	6	7	
Total Marks Including choice			95

PR GOVT COLLEGE (A):: KAKINADA
DEPARTMENT OF COMPUTER APPLICATIONS
III B.Sc (IT) SEMESTER-V
Course 15: Mobile Application Development

Time: 2 ½ Hrs

Max. Marks: 50

SECTION –I

Answer any three of the following questions. Must attempt at least one question from each part.
Each question carries 10 Marks. 3 X 10 = 30M

Part-A

1. Explain the evolution and history of mobile application frameworks and tools. Explain History of Mobile Application Framework and Tools?
2. Explain the evolution and history of mobile application frameworks and tools.
3. Write About Flutter Tool and explain Hot Reload and Hot Restart.

Part-B

4. Explain how Flutter widgets respond to state changes in declarative UI programming.
5. How do you use the Dart package manager to include packages in a Flutter project?
6. Describe the best practices in mobile application development management.

SECTION - II

Answer any four of the following questions. Each question carries 5 marks. 4 X 5 = 20M

7. Define 1-tier, 2-tier, and 3-tier architecture.
8. What are the main characteristics of mobile applications?
9. What is the purpose of the pubspec.yaml file in Flutter?
10. Mention any two layout widgets in Flutter.
11. What is the purpose of the setState() method in Flutter?
12. Mention one advantage of using Dart constructors in custom widgets.
13. What are mobile client security concerns?

P.R. GOVT. COLLEGE (A), KAKINADA
III B.Sc (IT) (W.E.F 2025-2026)
SEMESTER – III
QUESTION BANK

Subject: Mobile Application Development

Paper: 15

Essay Questions

Unit-1

1.	Explain the types of connections and synchronization used in mobile applications.	BT1	CO1	PO1
2.	Explain the evolution and history of mobile application frameworks and tools.	BT2	CO2	PO2
3.	Describe about Frame work and Tools For Mobile App Development?	BT2	CO3	PO3

Unit-2

1.	Write About Flutter Tool and explain Hot Reload and Hot Restart.	BT1	CO1	PO1
2.	Discuss how Material icons are incorporated into Flutter applications	BT2	CO2	PO2
3.	Explain the build() method and how layout widgets like Columns, Rows, Containers, and Cards are used.	BT2	CO3	PO3

Unit-3

1.	Explain how Flutter widgets respond to state changes in declarative UI programming.	BT1	CO1	PO1
2.	Explain the relationship between setState(), State objects, and Stateful widgets with an example.	BT2	CO2	PO2
3.	Discuss the use of Dart libraries, variables, and functions in Flutter apps.	BT2	CO3	PO3

Unit-4

1.	How do you use the Dart package manager to include packages in a Flutter project?	BT1	CO1	PO1
2.	Describe how to use Dart constructors to create customizable widgets.	BT2	CO2	PO2

Unit-5

1.	Discuss user-to-mobile client, mobile client, and client-server communication security issues.	BT1	CO1	PO1
2.	Explain the existing web architecture and backend system security issues in mobile development.	BT2	CO2	PO2

Short Questions:**Unit-1**

1.	Define 1-tier, 2-tier, and 3-tier architecture.	BT1	CO1	PO1
2.	What are the main characteristics of mobile applications?	BT2	CO2	PO2
3.	Name the components of a mobile device.	BT2	CO3	PO3

Unit-2

1.	What is the purpose of the pubspec.yaml file in Flutter?	BT1	CO1	PO1
2.	Mention any two layout widgets in Flutter.	BT2	CO2	PO2
3.	How do you use the Icons class in Flutter?	BT2	CO3	PO3

Unit-3

1.	Differentiate between Stateful and Stateless widgets.	BT1	CO1	PO1
2.	What is the purpose of the setState() method in Flutter?	BT2	CO2	PO2
3.	List basic data types used in Dart.	BT2	CO3	PO3

Unit-4

1.	What is the Dart package manager?	BT1	CO1	PO1
2.	What does "UI as code" mean in Flutter?	BT1	CO1	PO1
3.	Mention one advantage of using Dart constructors in custom widgets.	BT2	CO2	PO2

Unit-5

1.	List any two user-to-mobile client security issues.	BT1	CO1	PO1
2.	Define client-server communication security in mobile apps.	BT2	CO2	PO2
3.	What are mobile client security concerns?	BT2	CO2	PO2