SACRED HEART COLLEGE (AUTONOMOUS), THEVARA

KOCHI, KERALA, 682013

CURRICULUM AND SYLLABUS

FOR

BACHELOR OF COMPUTER APPLICATIONS (MOBILE APPLICATIONS AND CLOUD TECHNOLOGY)

CHOICE BASED CREDIT AND SEMESTER SYSTEM (CBCSS)

INTRODUCED FROM 2023 ADMISSIONS ONWARDS

Prepared by
Board of Studies in Computer Science
Sacred Heart College Thevara, Kochi.

SACRED HEART COLLEGE AUTONOMOUS, THEVARA, KOCHI-682013

Board of Studies in Computer Science 2021-24

Sl. No	Name & Address	Designation/Category
1	Dr. Regitha M R Asst. Professor & HoD Dept. of Computer Science Sacred Heart College, Thevara	Chairman
2	Rev. Dr. Jaison Paul Mulerikkal Vice Principal Rajagiri School of Engineering and Technology, Kakkanad	Expert in the subject from outside the college, nominated by the academic council.
3	Dr. Judy M V Professor Dept. of Computer Applications CUSAT	Expert in the subject from outside the college, nominated by the academic council.
4	Dr. Swaraj K.P., Associate Professor, Department of Computer Science & Engineering, Govt. College of Engineering, Thrissur	Expert nominated by the Vice Chancellor
5	Mr. Maxon C J Infrastructure Associative UST Global, Kochi	Post Graduate meritorious alumnus nominated by the Principal
6	Mr. Mathew James Azure Cloud Solution Architect UST Global, Kochi	Representative from industry, corporate – sector or allied area.
7	Ms. Tressa Shybe	Faculty member
8	Ms. Achamma Cherian	Faculty member
9	Ms. Jisha Soman	Faculty member
10	Ms. Rensi K Ranjith	Faculty member
11	Rev. Fr. Nijo Antony	Faculty member
12	Ms. Christy Jacqueline	Faculty member
13	Ms. Neethu A S	Faculty member
14	Mr. Santhosh Kumar K P	Faculty member

CONTENTS

1. INTRODUCTION 1

1.1 PREAMBLE 1

1.2 ELIGIBLITY: 1

1.3 OBJECTIVES: 1

1.4 LEARNING OUTCOMES 2

1.5 PROGRAMME OUTCOMES (PO) 2

1.6 PROGRAMME SPECIFIC OUTCOMES (PSO) 3

1.7 JOB OPPORTUNITIES: 3

2. REGULATIONS FOR CHOICE BASED CREDIT SEMESTER SYSTEM (CBCSS) 5

2.1. TITLE 5

2.2. SCOPE 5

2.3. DEFINITIONS 5

2.4. ATTENDANCE 8

2.5. PROGRAMME REGISTRATION 9

2.6. PROMOTION: 9

2.7. UNDER GRADUATE PROGRAMME STRUCTURE - Model III 10

2.8. SCHEME OF PROGRAMME 11

2.9. EXAMINATIONS 14

2.10. EVALUATION AND GRADING 14

2.11. REGISTRATION FOR THE EXAMINATION 19

2.12. SUPPLEMENTARY EXAMINATIONS 19

2.13. IMPROVEMENT OF EXAMINATION 19

2.14. PROMOTION TO THE NEXT HIGHER SEMESTER 19

2.15. CERTIFICATES 20

2.16. MONITORING 20

2.17. GRIEVANCE REDRESSAL MECHANISM 20

3. SYLLABUS 21

3.1 SEMESTER I 21

3.2 SEMESTER II 35

3.3 SEMESTER III 52

3.4 SEMESTER IV 71

3.5 SEMESTR V 89

3.6 SEMESTER VI 107

4. OPEN COURSE 120

5. MODEL QUESTION PAPERS 122

1. INTRODUCTION

1.1 PREAMBLE

The Department of Computer Science of Sacred Heart College started a programme on BCA (Mobile Applications and Cloud Technology) on 2016. As Sacred Heart College is granted academic autonomy in the year 2014 it has the privilege of restructuring the syllabus and introducing new career-oriented industry ready programs. Keeping an eye on the industry and to modernize the curriculum, the Board of Studies of the Department of Computer Science, Sacred Heart College, has initiated an industry a programme Bachelor of Computer Applications (Mobile Applications and Cloud Technology). This programme comprises complementary course **Mathematics,** and vocational courses **Mobile Applications, Cloud Technology.**

The main objective of this programme is to inculcate and horn up the skills of young minds in new generation technologies to compete in the knowledge era. The curriculum is designed to develop skills in android application development and cloud technology so as to enable the graduates to take up employment/self-employment in global technical market. This unique programme provides dual career options for the students in the latest and fast-growing technology sectors of **Mobile Applications and Cloud Technology.**

The syllabus aims to comprehend the theoretical foundations of computer science in modelling and developing solutions to the complex and real-world problems and apply the knowledge of layered network models, protocols, technologies, topologies and security policies for building network and internet-based applications.

The graduates are able to comprehend, explore and build up computer programs, applications and computer-based systems that meet the needs of industry and society.

1.2 ELIGIBLITY:

Candidates should have passed the Plus Two / equivalent examination with Mathematics/ Statistics/ Computer Science or Science with Informatics Practice as one of the optional subjects.

1.3 OBJECTIVES:

Provide a strong foundation in fundamentals of computers science. Prepare the students with exceptional skills of problem solving, communication and leadership skills. Facilitate knowledge, technologies and their applications through the different courses of the programme. Prepare the students to provide professional solutions to real time problems. In addition to all the mandatory core courses of BCA, this programme offers in-depth practical knowledge of the latest technology trends in Mobile Applications and Cloud Technology.

The vocational courses on Mobile Applications provides the fundamental knowledge on mobile technologies and android application development using Java, Kotlin, React, Flutter which meets industry standards.

The vocational courses on cloud technology provides the fundamental knowledge on virtualization technology, cloud technology, datacenters, DevOps, network administration using windows and Linux, and server operating systems.

1.4 LEARNING OUTCOMES

On completion of the Bachelor of Computer Applications in Mobile Applications programme, students should be able to:

- Use a range of programming languages and tools to develop computer programs and systems that are effective solutions to problems.
- Design precise specifications of algorithms, procedures, and interaction behavior.
- Apply mathematics, logic, and statistics to the design, development, and analysis of software systems.
- Equipped with a range of fundamental principles of Computer Science that will provide the basis for future learning and enable them to adapt to the constant rapid development of the field.
- Experience of working in teams to build software systems.

1.5 PROGRAMME OUTCOMES (PO)

PO 1: Critical Thinking: Take informed actions after identifying the assumptions that frame our thinking and actions, checking out the degree to which these assumptions are accurate and valid, and looking at our ideas and decisions (intellectual, organizational, and personal) from different perspectives.

PO 2: Effective Communication: Speak, read, write and listen clearly in person and through electronic media in English and in one Indian language, and make meaning of the word by connecting people, ideas, books, media and technology.

PO 3: Effective Citizenship: Demonstrate empathetic social concern and equity centered national development, and the ability to act an informed awareness of issues and participate in civic life through volunteering.

PO 4: Environment and Sustainability: Understand the issues of environmental contexts and sustainable development.

PO 5: Ethics: Recognize different value systems including your own, understand the moral dimensions of your decisions, and accept responsibility for them.

PO 6: Global Perspective: Understand the economic, social and ecological connections that link the world's nations and people.

1.6 PROGRAMME SPECIFIC OUTCOMES (PSO)

PSO 1: Apply the theoretical foundations of computer science in modelling and developing solutions to the complex and real-world problems.

PSO 2: Comprehend, explore and build up computer programs, applications in the allied areas like Algorithms, Multimedia, Web Design and android applications for efficient design of computer-based systems that meet the needs of industry and society

PSO 3: Develop skills in android and cloud technology development so as to enable the graduates to take up employment/self-employment in global technical market.

PSO 4: Apply knowledge of layered network models, protocols, technologies, topologies and security policies for building network and internet-based applications.

1.7 JOB OPPORTUNITIES:

Job Opportunities for Mobile App Developers

- Small, Medium and Large Professional Services IT Companies
- Enterprise Application Product and Service Companies
- Mobile Application Product and Service Companies
- VAS providers
- e-Commerce, m-Commerce companies

Career Progression Path - Mobile Applications

Industry	Entry level (0-1 yrs exp.)	Mid Level (3-5 yrs exp.)	Advanced level (5yrs plus exp.)
Job Role	UI Engineer	UI Designer	UI Development Lead
	Software Engineer / Developer	Sr. Software Engineer / Developer	Technical Lead / Technical Manager / Project Manager
	Test Engineer	Sr. Test Engineer	Test Manager
	Mobile Application Developer	Mobile Application Entrepreneur	CEO

Job Opportunities for Cloud Technology Professionals

By the enhancement of cloud technology students can obtain the opportunities as:

- Cloud architect in various companies
- Cloud infrastructure analyst
- Cloud sales specialist/ cloud brokerage
- Helps to work in cloud computing
- DevOps environment.
- Cloud support engineer Cloud administrator

Career Progression Path - Cloud Technology

Industry	Entry level (0-1 yrs exp.)	Mid-Level (3-5 yrs exp.)	Advanced level (5 yrs plus exp.)
Job Role	Cloud Architect	Sr. Cloud Architect	Cloud Consultant
	Cloud Engineer	Sr. Cloud Engineer Manager Cloud Technology	Manager Cloud Technology
	Datacenter Technician	Datacenter Engineer	Datacenter Manager
	Remote Desktop Engineer	Cloud Provisioning Engineer	Datacenter Manager
	Cloud Security Specialist	Security Engineer	Manager Cloud Security

2. REGULATIONS FOR CHOICE BASED CREDIT SEMESTER SYSTEM (CBCSS)

FOR UNDER GRADUATE PROGRAMMES 2023

2.1. TITLE

These regulations shall be called **“SACRED HEART COLLEGE THEVARA REGULATIONS FOR CREDIT AND SEMESTER SYSTEM 2023”**

2.2. SCOPE

Applicable to all programmes of the college with effect from 2019 admissions, except otherwise approved by the Academic Council of the College

2.3. DEFINITIONS

i. ‘**Programme**’ means the entire course of study and examinations.

ii. ‘**Duration of Programme**’ means the period of time required for the conduct of the programme. The duration of under graduate programmes shall be 6 semesters, post-graduate programme shall be of 4 semesters and M Phil programmes shall be 2 semesters.

iii. ‘**Semester**’ means a term consisting of a minimum of 90 working days, inclusive of examination, distributed over a minimum of 18 weeks of 5 working days, each with 5 contact hours of one hour duration

iv. ‘**Course**’ means a segment of subject matter to be covered in a semester. Each Course is to be designed variously under lectures / tutorials / laboratory or fieldwork/study tour/seminar / project / practical training / assignments/evaluation etc., to meet effective teaching and learning needs.

v. ‘**Common Course I**’ means a course that comes under the category of courses for English and ‘**Common Course II**’ means additional language, a selection of both is compulsory for all students undergoing undergraduate programmes.

vi. ‘**Core course**’ means a course in the subject within a degree programme.

vii. ‘**Complementary Course**’ means a course which would enrich the study of core courses.

viii. ‘**Open course**’ means a course outside the field of his/her specialization, which can be opted by a student.

ix. ‘**Additional core course**’ means a compulsory course for all under graduate students (as per the UGC directive) to enrich their general awareness.

x. The U.G. programmes shall include (a) Common courses (b) Core courses (c) Complementary Courses (d) Open Course (e) Study tour and (f) Internship for selected programmes.

xi. '**Additional Course**' is a course registered by a student over and above the minimum required courses.

xii. '**Credit**' (Cr) of a course is the numerical value assigned to a course according to the relative importance of the content of the syllabus of the programme.

xiii. '**Extra credits**' are additional credits awarded to a student over and above the minimum credits required for a programme for achievements in co-curricular activities carried out outside the regular class hours OR curricular activities/courses completed for value addition, as directed by the College/ department. It is the numerical value assigned to Club activities, Social service, Internship etc. which is not added with the total academic credits of the students. Additional credit components

a) Talent & career club activity (optional)

b) Social service (mandatory)

c) Internship for Commerce, Communication and Computer applications (mandatory).

d) Internship (desirable for other programmes).

e) Add on courses (optional)

xiv. '**Programme Credit**' means the total credits of the UG Programme.

xv. '**Programme Elective course'** Programme Elective course means a course, which can be chosen from a list of electives and a minimum number of courses is required to complete the programme.

xvi. '**Programme Project'** Programme Project means a regular project work with stated credits on which the student undergoes a project under the supervision of a teacher in the parent department / any appropriate Institute in order to submit a dissertation on the project work as specified.

xvii. **'Internship'** is on-the-job training for professional careers.

xviii. **'Plagiarism'** Plagiarism is the unreferenced use of other authors' material in dissertations and is a serious academic offence.

xix. **'Tutorial'** Tutorial means a class to provide an opportunity to interact with students at their individual level to identify the strength and weakness of individual students.

xx. **‘Seminar’** means a lecture by a student expected to train the student in self- study, collection of relevant matter from the books and Internet resources, editing, document writing, typing and presentation.

xxi. **‘Evaluation’** means every course shall be evaluated by 25% continuous (internal) assessment and 75% end course/end semester (external) assessment.

xxii. **‘Repeat course’** is a course that is repeated by a student for having failed in that course in an earlier registration.

xxiii. **‘Audit Course’** is a course for which no credits are awarded.

xxiv. **‘Department’** means any teaching Department offering a course of study approved by the college / Institute as per the Act or Statute of the University.

xxv. **‘Parent Department’** means the Department which offers a particular UG/PG programme.

xxvi. **‘Department Council’** means the body of all teachers of a Department in a College.

xxvii. **‘Faculty Advisor’** is a teacher nominated by a Department Council to coordinate the continuous evaluation and other academic activities undertaken in the Department.

xxviii. **‘College Co-ordinator**’ means a teacher from the college nominated by the College Council to look into the matters relating to CBCS-PG System.

xxix. **‘Letter Grade’** or simply **‘Grade’** in a course is a letter symbol (O, A, B, C, D, etc.) which indicates the broad level of performance of a student in a course.

xxx. Each letter grade is assigned a ‘**Grade point**’ (GP) which is an integer indicating the numerical equivalent of the broad level of performance of a student in a course.

xxxi. **‘Credit point’** (CP) of a course is the value obtained by multiplying the grade point (GP) by the Credit (Cr) of the course CP=GP x Cr.

xxxii. **‘Semester Grade point average’** (SGPA) is the value obtained by dividing the sum of credit points (CP) obtained by a student in the various courses taken in a semester by the total number of credits taken by him/her in that semester. The grade points shall be rounded off to two decimal places. SGPA determines the overall performance of a student at the end of a semester.

xxxiii. ‘**Cumulative Grade point average**’ (CGPA) is the value obtained by dividing the sum of credit points in all the courses taken by the student for the entire programme by the total number of credits and shall be rounded off to two decimal places.

xxxiv. **‘Grace Marks’** means marks awarded to course/s, as per the orders issued by the college from time to time, in recognition of meritorious achievements in NCC/NSS/Sports/Arts and cultural activities.

2.4. ATTENDANCE

Being a regular college, physical presence in the regular activities, especially, classes and exams, is mandatory for the students. However, if a student secures 75% of attendance s/he is eligible to appear for the exams, provided there are no other impediments like disciplinary proceedings, malpractice record etc.

i. A maximum of 5 marks (5%) for a course is given for attendance

ii. **Absence**: A student found absent for one hour in the forenoon or afternoon session is deprived of the attendance for the entire session as far as eligibility for final exam is concerned.

iii. The hour related calculation in a course is meant for awarding marks for the course concerned.

iv. **Late entry:** A student is supposed to be in time in the class. Late arrival related treatment is left to the discretion of the individual teacher. However, as a norm, a late arriving student may be permitted to the class, if it is not inconvenient or distraction to the class as such; though attendance MAY NOT BE GIVEN. Late arrival beyond 5 minutes is treated as ABSENCE; though the teacher may consider permitting the student to sit in the class.

v. **Leave**: A student has to formally report his/her absence with reasons either in advance, or immediately after the absence for obtaining an approved leave. This applies to all sorts of leave – medical, on duty or other.

vi. The student is supposed to report in prescribed format on the very next day of the absence; however, up to a week’s time is permitted. Afterwards, the leave applications will not be considered.

vii. The student has to retain a copy/section of the approved leave form and produce the same as proof, in case there is any confusion regarding the leave sanctioning. In the absence of such proof, the claims will not be entertained.

viii. **Duty Leave:** A student representing the college in sports, arts, social service or academic matters, has to get sanction from the class teacher concerned and submit the leave application form duly endorsed by teacher concerned & the class teacher, and submit it to the faculty Dean (or Vice Principal). The same will be forwarded by the Dean/Vice Principal for attendance entry.

ix. **SPORTS**: The approval of the Department of Physical Education and the class teacher is required. The time limit for submission mentioned above is applicable in the case of duty leave as well.

x. **CONDONATION**: A student may have the privilege of condonation of attendance shortage (upto a maximum of 10 days) on the basis of genuineness of the grounds of absence (medical reasons or college duty), duly recommended by the department. This is not a matter of right. It is a matter of privilege based on Principal's discretion and the good conduct of the student on the campus. A student of UG programme may have a maximum of two such opportunities and that of PG programmes only one opportunity.

xi. **RE-ADMISSION:** A student whose attendance is inadequate will have to discontinue the studies. Such students, whose conduct is good, may be re-admitted with the approval of governing council, on the basis of recommendation from the department, and assurance from the student and the guardian regarding good conduct and compliance in academic and discipline matters. For this the prescribed re-admission fee has to be paid.

As a condition for re-admission, the student should have cleared all academic arrears, or should have appeared for the exams in which he/she is having an arrear (if the results are not out), and should have fulfilled all academic assignments prescribed by the department for compensating for his lack of attendance.

xii. **UNAUTHORISED ABSENCE & REMOVAL FROM ROLLS:** A student absent from the classes continuously for 10 consequent days without intimation or permission, shall be removed from the rolls, and the matter intimated to the student concerned. On the basis of recommendation of the department concerned, re-admission process may be permitted by the Principal.

2.5. PROGRAMME REGISTRATION

i. A student shall be permitted to register for the programme at the time of admission.

ii. A UG student who registered for the programme shall complete the same within a period of 6 continuous semesters and a PG student within a period of 4 continuous semesters from the date of commencement of the programme.

2.6. PROMOTION:

A student who registers for the end semester examination shall be promoted to the next semester. However, in extreme circumstances, a student having sufficient attendance who could not register for the end semester examination may be allowed to register notionally by the Principal with the recommendation of the Head of the Department concerned and, by paying the prescribed fee.

2.7. UNDER GRADUATE PROGRAMME STRUCTURE - Model III

A	Programme Duration	6 Semesters
B	Total Credits required for successful completion of the Programme	120
C	Total Credits for Common course	8
D	Total Credits for Complementary Course	8
E	Credits for Core Courses	62
F	Credits for Vocational Courses: Mobile applications	16
G	Credits for Vocational Courses: Cloud Technology	23
H	Open Course	3
I	Minimum attendance required	75%

2.8. SCHEME OF PROGRAMME

SEMESTER 1							
COURSE CODE	COURSE TITLE	COURSE TYPE	COURSE CATEGORY	COURSE STREAM	NO. HRS. /WEEK	CREDITS	TOTAL HRS./SEM
23U1CCENG1	Homo Loquens: Effective Listening and Speaking	Theory	Common Course	English	5	4	90
23U1CPCMT1	Foundation of mathematics	Theory	Complementary	Mathematics	4	4	72
23U1CRBCA1	Digital System Design	Theory	Core Course	Computer Applications	4	4	72
23U1CRBCA2	Introduction to Programming	Theory	Core Course	Computer Applications	4	3	72
23U1CRBCA3	Computer organization	Theory	Core Course	Computer Applications	4	3	72
23U1PRBCA1	Introduction to Programming-Practical	Practical	Core Course	Computer Applications	2	1	36
23U1PRBCA2	Digital System Design and Linux -Practical	Practical	Core Course	Computer Applications	2	1	36
Total					**25**	**20**	**450**
SEMESTER 2							
23U2CCENG03	Text and Context: A Guide to Effective Reading and Writing	Theory	Common Course	English	5	4	90
23U2CPCMT2	Discrete Mathematics	Theory	Complementary	Mathematics	4	4	72
23U2CRBCA4	Operating system	Theory	Core Course	Computer Applications	4	3	72
23U2CRBCA5	Problem solving using Python	Theory	Core Course	Computer Applications	4	3	72
23U2CRBCA6	Data structures and algorithms using C++	Theory	Core Course	Computer Applications	4	3	72
23U2PRBCA3	Problem solving using Python-Practical	Practical	Core Course	Computer Applications	2	1	36
23U2PRBCA4	Data structures and algorithms using C++-Practical	Practical	Core Course	Computer Applications	2	1	36
Total					**25**	**19**	**450**

SEMESTER 3							
COURSE CODE	**COURSE TITLE**	**COURSE TYPE**	**COURSE CATEGORY**	**COURSE STREAM**	**NO. HRS. /WEEK**	**CREDITS**	**TOTAL HRS./SEM**
23U3CRBCA7	Basic Statistics	Theory	Core Course	Computer Applications	4	4	72
23U3CRBCA8	Software engineering	Theory	Core Course	Computer Applications	4	3	72
23U3CRBCA9	RDBMS	Theory	Core Course	Computer Applications	4	4	72
23U3CRBCA10	Computer networks	Theory	Core Course	Computer Applications	5	4	90
23U3CRBCA11	Programming in JAVA	Theory	Core Course	Computer Applications	4	3	72
23U3PRBCA5	RDBMS – Practical	Practical	Core Course	Computer Applications	2	1	36
23U3PRBCA6	Programming in JAVA – Practical	Practical	Core Course	Computer Applications	2	1	36
Total					**25**	**20**	**450**
SEMESTER 4							
23U4CRBCA12	Web Technologies	Theory	Core Course	Computer Applications	4	4	72
23U4VCBCA1	Introduction to Cloud Technology	Theory	Vocational Course	Cloud Technology	4	4	72
23U4VCBCA2	Fundamentals of Data Centre and storage	Theory	Vocational Course	Cloud Technology	4	4	72
23U4VCBCA3	Android Application Development with Kotlin	Theory	Vocational Course	Mobile Applications	4	3	72
23U4CRBCA13	Mobile Device and Network Architecture	Theory	Core Course	Computer Applications	4	3	72
23U4PVBCA1	Android Application Development with Kotlin – Practical	Practical	Vocational Course	Mobile Applications	3	2	54
23U4PVBCA2	Introduction to Cloud Technology – Practical	Practical	Vocational Course	Cloud Technology	2	1	36
Total					**25**	**21**	**450**

SEMESTER 5							
COURSE CODE	**COURSE TITLE**	**COURSE TYPE**	**COURSE CATEGORY**	**COURSE STREAM**	**NO. HRS. /WEEK**	**CREDITS**	**TOTAL HRS./SEM**
23U5VCBCA4	Principles of Virtualization	Theory	Vocational Course	Cloud Technology	4	4	72
23U5VCBCA5	Server Operating System	Theory	Vocational Course	Cloud Technology	4	4	72
23U5CRBCA14	IT and Environment	Theory	Core Course	Computer Applications	4	4	72
	Open Course	Theory	Open Course	Computer Applications	4	3	72
23U5VCBCA6	Hybrid Mobile Application Development	Theory	Vocational Course	Mobile Applications	4	3	72
23U5PVBCA3	Server Operating System – Practical	Practical	Vocational Course	Cloud Technology	2	1	36
23U5PVBCA4	Hybrid Mobile Application Development -Practical	Practical	Vocational Course	Mobile Applications	3	2	54
Total					**25**	**21**	**450**
SEMESTER 6							
23U6VCBCA7	DevOps Fundamentals	Theory	Vocational Course	Cloud Technology	3	3	54
23U6CRBCA15	Introduction to Machine Learning and Data science	Theory	Core Course	Computer Applications	4	4	72
23U6CRBCA16	Computer Graphics using OpenGL	Theory	Core Course	Computer Applications	3	3	54
23U6PVBCA5	Software Lab	Practical	Vocational Course	Cloud Technology	3	2	54
23U6PRBCA7	Introduction to Machine Learning and Data science -Practical	Practical	Core Course	Computer Applications	2	1	36
23U6PJBCA1	Project and Viva-Voce	Practical	Vocational Course	Mobile Applications	10	6	180
Total					**25**	**19**	**450**

2.9. EXAMINATIONS

All the End Semester Examinations of the college will be conducted by the Controller of Examination. The Principal will be the Chief Controller of Examinations. An Examination committee consists of the Chief Controller of Examinations, Controller of Examinations, Additional Chief Superintendent, Deans, IQAC Coordinator and other faculty members nominated by the Principal will act as an advisory body of the matters relating to the conduct of examinations.

2.10. EVALUATION AND GRADING

The evaluation scheme for each course shall contain two parts;

a. Continuous Internal Evaluation (CIA) and
b. End Semester Examination (ESE).

The internal to external assessment ratio shall be 1:3, for both courses with or without practical. For courses without practical, there shall be a maximum of 75 marks for external evaluation and maximum of 25 marks for internal evaluation. For courses with practical, generally external evaluation shall be for a maximum of 60 marks and internal evaluation for 20 marks. Both internal and external evaluation shall be carried out in the mark system and the marks are to be rounded to the nearest integer.

2.10.1. Continuous Internal Assessment (CIA)/ Continuous Assessment:

The internal evaluation shall be based on predetermined transparent system involving periodic written tests, assignments, seminars/viva/field survey and attendance in respect of theory courses and based on written tests, lab skill/records/viva and attendance in respect of practical courses. The marks assigned to various components for internal evaluation as follows

Components of Internal Evaluation for theory courses

	Components	**Marks**
i.	Assignments	5
ii	Seminar/Quiz/Field survey /Viva etc.	5
iii	Attendance	5
iv	Two Test papers(2x5)	10
	Total	25

i. **Assignments**: Every student shall submit one assignment as an internal component for every course.

Components	**Marks**
Punctuality	1
Content	2
Conclusion	1
Reference/Review	1
Total	**5**

ii. **Seminar**: The seminar lecture is expected to train the student in self-study, collection of relevant matter from the books and Internet resources, editing, document writing, typing and presentation.

Components	**Marks**
Content	2
Presentation	2
Reference/Review	1
Total	**5**

iii. Evaluation of Attendance

The attendance of students for each course shall be another component of internal assessment.

% of attendance	**Mark**
Above 90%	5
Between 85 and below 90	4
Between 80 and below 85	3
Between 76 and below 80	2
Between 75 and below 76	1

Components of Internal Evaluation for Practical Courses

Components of Theory – Internal Evaluation	Marks
Attendance	5
Continuous internal laboratory assessment	20
Total	25

Components of Continuous internal lab assessment

The continuous assessment of laboratory experiments will be evaluated by the following components:

Components	Marks
Lab involvement/Performance	12
Record	3
Mini project/Viva	5
Total	20

iv. **Class Tests:** Every student shall undergo two class tests as an internal component for every course.

2.10.2. End Semester Examination (ESE):

The End Semester Examination in theory courses shall be conducted by the college with question papers set by external experts/ question bank. The evaluation of the answer scripts shall be done by the examiners based on a well-defined scheme of evaluation given by the question paper setters/Prepared as per the direction of the Chairman, Board of Examiners. The evaluation of the End Semester Examinations shall be done immediately after the examination preferably through the centralised valuation.

2.10.3. Project

Project work is a part of the syllabus of most of the programmes offered by the college. The guidelines for doing projects are as follows:

i. Project work shall be completed by working outside the regular teaching hours.

ii. Project work shall be carried out under the supervision of a teacher in the concerned department or an external supervisor.

iii. A candidate may, however, in certain cases be permitted to work on the project in an industrial / Research Organization/ Institute on the recommendation of the Supervisor.

iv. There should be an internal assessment and external assessment for the project work in the ratio 2:3

v. The external evaluation of the project work consists of valuation of the dissertation (project report) followed by presentation of the work and viva voce.

vi. The mark and credit with grade awarded for the program project should be entered in the grade card issued by the college.

Components of Internal Evaluation for Projects

Components	**Marks**
Topic of relevance	2
Requirement analysis and system study	10
Intermediate Presentation	10
Demo	10
Viva	5
Punctuality-Regularity	3
Total	**40**

2.10.4. Comprehensive Viva-voce

Comprehensive Viva-voce shall be conducted during project viva which covers questions from all courses in the programme as per the syllabus.

2.10.5. Grade and Grade Points

For all courses (theory & practical), Letter grades and grade point are given on a 10-point scale based on the total percentage of marks, (CIA+ESE) as given below:-

Percentage of Marks	Grade	Grade Point (GP)
95 and above	S Outstanding	10
85 to below 95	A^+ Excellent	9
75 to below 85	A Very Good	8
65 to below 75	B^+ Good	7
55 to below 65	B Above Average	6
45 to below 55	C Average	5
35 to below 45	D Pass	4
Below 35	F Fail	0
	Ab Absent	0

Grades for the different semesters and overall programme are given based on the corresponding SGPA/CGPA as shown below:

SGPA/CGPA	Grade
Equal to 9.5 and above	**S OutStanding**
Equal to 8.5 and below 9.5	**A+ Excellent**
Equal to 7.5 and below 8.5	**A Very Good**
Equal to 6.5 and below 7.5	**B+ Good**
Equal to 5.5 and below 6.5	**B Above Average**
Equal to 4.5 and below 5.5	**C Average**
Equal to 3.5 and below 4.5	**D Pass**
Below 3.5	**F Failure**

A separate minimum of 30% marks each for internal and external (for both theory and practical) and aggregate minimum of 35% are required for a pass for a Under Graduate programme. A candidate who has not secured minimum marks/credits in internal examinations can re-do the same registering along with the end semester examination for the same semester, subsequently. A student who fails to secure a minimum marks/grade for a -pass in a course can be permitted to write the examination along with the next batch.

After the successful completion of a semester, Semester Grade Point Average (SGPA) of a student in that semester is calculated using the formula given below. For the successful completion of semester, a student should pass all courses and score at least the minimum CGPA grade 'D'. However, a student is permitted to move to the next semester irrespective of her/his SGPA.

Credit Point (CP) of a course is calculated using the formula

CP = Cr x GP, where Cr = Credit; GP = Grade point

Semester Grade Point Average (SGPA) of a Semester is calculated using the formula

SGPA = TCP/TCr, where

TCP = Total Credit Point of that semester $= \sum_{1}^{n} \mathbf{CPi}$;

TCr = Total Credit of that semester $= \sum_{1}^{n} \mathbf{Cri}$

Where n is the number of courses in that semester

Cumulative Grade Point Average (CGPA) of a Programme is calculated using the formula

CGPA = $\sum(\mathrm{SGPA} \times \mathrm{TCr}) \Big/ \sum \mathrm{TCr}$

SGPA/CGPA shall be round off to two decimal places

To ensure transparency of the evaluation process, the internal assessment marks awarded to the students in each course in a semester shall be published on the notice board/website at least one week before the commencement of external examination. There shall not be any chance for improvement for internal mark.

The course teacher and the faculty advisor shall maintain the academic record of each student registered for the course which shall be forwarded to the controller of examinations through the Head of the Department and a copy should be kept in the department for at least two years for verification.

2.11. REGISTRATION FOR THE EXAMINATION

a. All students admitted in a programme with remittance of prescribed fee are eligible for the forthcoming semester examinations.

b. Online application for registration to the various End Semester Examinations shall be forwarded to the Controller of Examination along with prescribed fee for each course in prescribed format.

c. The eligible candidates who secure the prescribed minimum attendance of the total duration of the course and possess other minimum qualification prescribed in the regulations for each course shall be issued the hall tickets. The hall ticket shall be downloaded by the students from the college website.

d. The mode of fee remittance shall be through the prescribed bank.

2.12. SUPPLEMENTARY EXAMINATIONS

Candidates who failed in an examination can write the supplementary examination conducted by the College along with regular examinations.

2.13. IMPROVEMENT OF EXAMINATION

A candidate can improve his/her marks once by appearing again for the examination with the subsequent batch with the remittance of prescribed fee. In such cases the better of the two marks shall be taken as the marks awarded to him.

Internal assessment marks shall be carried over to the subsequent semester examination. There shall not be any provision for improving internal assessment marks.

2.14. PROMOTION TO THE NEXT HIGHER SEMESTER

A candidate shall be eligible for promotion from one semester to the next higher semester if,

a. He / she secures a minimum 75 % attendance and registered for the End Semester Examination of the programme for which he/she is studying.

b. His / her progress of study and conduct are satisfactory during the semester completed, as per the assessments recorded by the course teachers and the Head of the Department concerned.

2.15. CERTIFICATES

1. Diploma and Degree certificates are issued by the Mahatma Gandhi University, Kottayam as per the act and statues of the University on the submission of the consolidated mark / score cards of the students by the College.
2. A consolidated mark / scored card shall be issued to the candidates after the publication of the results of the final semester examination taken by the candidate.
3. A Course Completion Certificate with classification shall be issued to students till the provisional certificate is issued by the university.

2.16. MONITORING

There shall be a Monitoring Committee constituted by the principal consisting of faculty advisors, Head of Department, a member from teaching learning evaluation committee (TLE) and the Deans to monitor the internal evaluations conducted by college. The Course teacher, Class teacher and the Deans should keep all the records of the internal evaluation, for at least a period of two years, for verification.

Every Programme conducted under Choice Based Credit System shall be monitored by the College Council under the guidance of IQAC Coordinator, Controller of Exams, academic deans and Head of the Department.

2.17. GRIEVANCE REDRESSAL MECHANISM

In order to address the grievance of students regarding Continuous internal assessment (CIA) a three-level Grievance Redressal mechanism is envisaged. A student can approach the upper level only if grievance is not addressed at the lower level.

Level 1: At the level of the concerned course teacher

Level 2: At the level of a department committee consisting of the Head of the Department, a coordinator of internal assessment for each programme nominated by the Head of the Department and the course teacher concerned.

Level 3: A committee with the Principal as Chairman, Dean of the Faculty concerned, Head of the Department concerned and one member of the Academic council nominated by the principal every year as members.

3. SYLLABUS

3.1 SEMESTER I

HOMO LOQUENS: EFFECTIVE LISTENING AND SPEAKING

Course Code	23U1CCENG1
No. of credits	4
No. of contact hours	90

Course Description: To develop the student's ability to use English language accurately and effectively by enhancing their communication skills.

Course Outcomes (COs)

CO1	Understand the speech sounds of English in order to enable them to listen to English and speak with global intelligibility.
CO2	Equip the students to speak English confidently and effectively in a wide variety of situations.
CO3	Improve the reading efficiency by refining their reading strategies.

Module 1: Speech Sounds (18 hours)

Phonemic symbols - Vowels - Consonants - Syllables - Word stress - Stress in polysyllabic words – Stress in words used as different parts of speech - Sentence stress – Weak forms and strong forms – Intonation – Awareness of different accents: American, British and Indian – Influence of the mother tongue

Module II: Listening (18 hours)

Active listening – Barriers to listening – Listening and note taking– Listening to announcements – Listening to news on the radio and television

Module III: Speaking (36 hours)

Word stress and rhythm – Pauses and sense groups – Falling and rising tones –Fluency and pace of delivery – Art of small talk – Participating in conversations – Making a short formal speech – Describing people, place, events and things – Group discussion skills and telephone skills

Module IV: Reading **(18 hours)**

Reading: theory and Practice – Scanning - Surveying a textbook using an index - reading with a purpose – making predictions – Understanding text structure – Locating main points – Making inferences - Reading graphics - reading critically – Reading for research

Text Book:

1. Sasikumar.V, Kiranmai Dutt P and Geetha Rajeevan, Communication Skills in English. Cambridge University Press and Mahatma Gandhi University.

SEMESTER I

FOUNDATIONS OF MATHEMATICS

Course Code	23U1CPCMT1
No. of credits	4
No. of contact hours	72

Course Description:

This course introduces the concepts of sets and functions, mathematical logic, and methods of proof. A brief introduction of theory of Numbers is also included. These topics are foundations of most areas of modern mathematics, and are applied frequently in the succeeding semesters.

Course Outcomes (COs)

CO1	Demonstrate a working knowledge of set notation and elementary set theory, recognize the connection between set operations and logic, properties of injections, surjections, bijections, compositions, and inverse functions
CO2	Determine equivalence relations, equivalence classes, partial ordering with the properties of set operations
CO3	Check the validity of predicates in Propositional and Quantified Propositional Logic using truth tables, deductive reasoning and inference theory on Propositional Logic
CO4	Interpret the concepts of divisibility, congruence, greatest common divisor, prime, and prime-factorization used in the area of cryptography

Module 1: Set theory: (15 hours)

Sets, set operations, functions, sequences and summations.

Module 2: Relations: (20 hours)

Relations and their properties, n-ary relations and their applications, representing relations, equivalence relations, partial orderings.

Module 3: Basic Logic (20 hours)

Propositional logic, Propositional equivalences, Predicates and quantifiers nested quantifiers, Rules of inference, Introduction to proofs, Proof methods and strategy.

Module 4: Theory of Numbers **(17 hours)**

Divisibility theory in the integers, the greatest common divisor, the Euclidean algorithm (division algorithm), Primes. The fundamental theorem of arithmetic. The theory of congruence. Basic properties of congruence. Fermat's little theorem Wilson's theorem. Euler's phi-function. Euler's generalization of Fermat's theorem.

Text Books:

1. K.H. Rosen: Discrete Mathematics and its Applications (Sixth edition), Tata McGraw Hill Publishing Company, New Delhi.
2. S. Bernard and J.M Child: Higher Algebra, AITBS Publishers, India.

References:

1. Lipschutz, Set Theory and related topics. 2nd ed, Schaum Outline Series, Tata McGraw-Hill Publishing Company, New Delhi.
2. P.R. Halmos Naive Set Theory, 1st ed, Springer-Verlag New York.
3. George E. Andrews, Number Theory. W.B. Saunders Company.
4. Ian Chiswell & Wifrid Hodges Mathematical Logic. Oxford university press.
5. Graham Everest and Thomas Ward, An Introduction to Number Theory. Springer, London.
6. Fernando Rodriguez Villegas, Experimental Number Theory, Oxford University Press.
7. Richard Johnsonbaugh Discrete Mathematics (Pearsons)
8. C.Y Hsiung Elementary Theory of Numbers, Allied Publishers

SEMESTER I

DIGITAL SYSTEM DESIGN

Course Code	23U1CRBCA1
No. of credits	4
No. of contact hours	72

Course Description:

The course focuses on designing combinational and sequential building blocks, using these building blocks to design bigger digital system

Course Outcomes (COs)

CO1	Interpret the Decimal, Binary, Octal, Hexadecimal number system conversions and the arithmetic operations involved in each number system
CO2	Simplify the Boolean expressions using K Map
CO3	Design of the combinational circuits adder, subtractor, encoder, decoder, multiplexer and demultiplex
CO4	Design of sequential circuits RS, JK, D, T and Master Slave
CO5	Design of shift registers and synchronous and asynchronous counters

Module 1: NUMBER SYSTEM (15 Hours)

Number Systems – Decimal, Binary, Octal, Hexadecimal - conversion from one system to another – Representation of negative numbers. Arithmetic Operations –Addition, Subtraction, Multiplication, Division of Binary numbers, Representation of floating-point numbers. Representation of BCD numbers, BCD Addition, Binary Codes (Addition and subtraction)) – Gray codes, excess 3 code- Character Coding Schemes – ASCII, EBCDIC.

Module 2: BOOLEAN ALGEBRA & LOGIC GATES (15 Hours)

Logic gates- AND, OR, NOT, NAND and NOR – Truth tables and graphical representation, Basic laws of Boolean Algebra, Simplification of Expressions, De Morgan’s theorems, Canonical expressions, Simplification of expression using K-MAP (up to 4 variables), Representation of simplified expressions using NAND/NOR Gates, Don’t care conditions

Module 3: COMBINATIONAL LOGIC (15 Hours)

Combinational Circuits – Analysis and Design Procedures - Binary Adder- Sub tractor (Half & Full) - Carry look ahead adder, BCD adder, code converter, XOR Applications Decoders, Encoders Parity bit Generator and checker, Multiplexers, DE multiplexers, Implementation of Boolean functions using MUX.

Module 4: SEQUENTIAL LOGIC CIRCUITS (15 Hours)

Sequential Circuits - Storage Elements: Latches, Flip-Flops – RS, JK, D, T and Master Slave, Triggering of flip-flops, Analysis of Clocked Sequential Circuits - Design Procedure-using JK, D & T.

Module 5: COUNTERS AND SHIFT REGISTERS (12 Hours)

Registers - Shift Registers – SISO, PIPO, SIPO, PISO- Universal shift registers, Counters-Design of Counters- Synchronous & Asynchronous Counters, BCD counter, Johnson counter, Ring counter

Text Books

1. Mano M.M Digital Logic and Computer design/Computer Architecture 1ST Edition-Pearson
2. Floyd Digital Electronics- Pearson/Prentice Hall

References

1. Charles H Roth, Jr, Lizy Kurian John, Digital System Design using VHDL,2/e, Cengage Learning
2. Mano M. M. and Michael D. Ciletti, Digital Design with an Introduction to the Verilog HDL, 5/e, Pearson Education, 2021.
3. Tokheim R. L., Digital Electronics Principles and Applications, 7/e, Tata McGraw Hill, 2007.
4. Rajaraman V. and T. Radhakrishnan, An Introduction to Digital Computer Design, 5/e, Prentice Hall India Private Limited, 2021.
5. Leach D, Malvino A P, Saha G, Digital Principles and Applications, 8/e, McGraw Hill Education, 2015.
6. M. Morris Mano, Computer System Architecture, 3/e, Pearson Education, 2021.
7. N. N. Biswas, "Minimization of Boolean Functions," in IEEE Transactions on Computers, vol. C-20, no. 8, pp. 925-929, Aug. 1971 10.1109/T-C.1971.223373

SEMESTER I

INTRODUCTION TO PROGRAMMING

Course Code	23U1CRBCA2
No. of credits	3
No. of contact hours	72

Course Description:

The course aims to introduce the field of programming and also gives an exposure to problem-solving through programming.

Course Outcomes (COs)

CO1	Interpret the memory organization of variables, arrays, structures and union in C
CO2	Implement the functions and pointers using C and C++
CO3	Interpret dynamic memory allocation and file handling concepts in C and C++
CO4	Summarize Object Oriented Programming concepts
CO5	Demonstrate inheritance and generics concepts with examples

Module 1: Fundamentals of C programming: (15 Hours)

Overview of C, Data Types, Constants & Variables, Operators & Expressions, Branching and Looping, **Arrays**- single & multidimensional arrays, **Basic I/O**-formatted and Unformatted I/O. **Structures**- Basics, declaring, referencing structure elements, array of structures, **Unions** – Declaration, uses, enumerated data-types, typedef.

Module 2: Bridging to C++- Pointers and functions with C and C++ (15 Hours)

Streams for input and output, reference variables, new data types-bool and wide characters, Variable declaration and scope. Additional/special operations in C++, const keyword. **Pointers**- The & and * operator, pointer expression, assignments, arithmetic, comparison, arrays of pointers, pointers to pointers, initializing pointers, pointers to functions, function returning pointers, **Functions**-fundamentals – general form, function arguments, parameter passing, recursion- basic concepts

Module 3: Dynamic memory allocation and file handling (15 Hours)

Memory layout of a program, dynamic memory allocation in C: malloc, calloc, realloc, free.

Dynamic memory allocation in C++: new, delete

File Handling in C: The file pointer, file accessing functions, fopen, fclose, fputc, fgetc, fprintf, fscanf. **File handling in C++:** opening and closing files using streams, ofstream, ifstream, fstream, tellg,seekg. Command line arguments

Module 4: Object oriented programming: (15 Hours)

Object oriented concepts overview: **class and objects**: Definition-defining the class, defining data members and member functions, constructors and destructors. Access specifier-private, public, protected. **Creating objects**: static and dynamic, handling array of objects. Friend function and inline-functions

Overloading: Function overloading, constructor overloading, operator overloading

Module 5: Inheritance and generics (12 Hours)

Inheritance: Base Class & derived class, defining derived classes, protected access specifier, public inheritance and private inheritance-member accessibility, Level of inheritance-single inheritance, multiple inheritance, multi-level inheritance, hierarchical inheritance, hybrid inheritance. Virtual base class and virtual functions. **Generic:** template classes and template functions.

Overview of: standard template libraries (STLs), visual C++: win32, mfc applications.

References:

1. Yashwant Kanetka Let us C, 6th Edition, PBP Publication.
2. Richie and Kenninghan, The C programming Language, BPB Publication.
3. Balaguruswamy, Programming in ANSI C, 3rd Edition, Tata McGraw Hill.

SEMESTER I

COMPUTER ORGANIZATION

Course Code	23U1CRBCA3
No. of credits	3
No. of contact hours	72

Course Description:

This course introduces the principles of computer organization and the basic architecture concepts. The course emphasizes performance and cost analysis, instruction set design, pipelining, memory technology, memory hierarchy, virtual memory management, and I/O systems.

Course Outcomes (COs)

CO1	Explain the basic computer components and its organization
CO2	Interpret the instructions, instruction format, addressing modes associated with CPU
CO3	Illustrate the concepts of memory, read only memory, random access memory and virtual memory
CO4	Explain the I/O concepts, DMA and characteristics of multiprocessors
CO5	Summarize the procs and cons of language translators used in programming language

Module 1 **(15 Hours)**

Basic computer organization and design: Functional Units of a computer, General architecture, Instruction codes, Computer Registers, Computer Instructions and Instruction cycle. Timing and Control, Memory-Reference Instructions, Input-Output and interrupt.

Module 2 **(15 Hours)**

Central processing unit: Stack organization, Instruction Formats, Addressing Modes-Implied / Implicit Addressing Mode, Stack Addressing Mode, Immediate Addressing Mode, Direct Addressing Mode, Indirect Addressing Mode, Register Direct Addressing Mode, Register Indirect Addressing Mode, Relative Addressing Mode, Indexed Addressing Mode, Base Register Addressing Mode, Data Transfer and Manipulation, Complex Instruction Set

Computer (CISC) Reduced Instruction Set Computer (RISC), CISC vs RISC

Module 3 **(15 Hours)**

Memory system: Memory Hierarchy, Semiconductor Memories, RAM (Random Access Memory), Read Only Memory (ROM), Types of ROM, Cache Memory, Performance considerations, Virtual memory, Paging, Secondary Storage, RAID.

Module 4 **(15 Hours)**

Input output: I/O interface, Programmed IO, Memory Mapped IO, Interrupt Driven IO, DMA. **Multiprocessors:** Characteristics of multiprocessors, Interconnection structures, InterProcessor Arbitration, Inter processor Communication, Cache Coherence.

Module 5 **(12 Hours)**

Language Translators: High level language, Low level language, Generation of programming languages, Assembler, Compiler, Interpreter, Advantages and disadvantages language translators

Text Books:

1. M. Morris Mano, Computer System Architecture, 3rd edition, Pearson/PHI, India.

References:

1. Carl Hamacher, Zvonks Vranesic, SafeaZaky, Computer Organization, 5th edition,
2. McGraw Hill, New Delhi, India.
3. William Stalling, Computer Organization and Architecture- designing for performance, 8th edition, Prentice Hall, New Jersey.
4. Anrew S. Tanenbaum, Structured Computer Organization, 5th edition, Pearson Education Inc,
5. John P. Hayes, Computer Architecture and Organization, 3rd edition, Tata McGrawHill

SEMESTER I

INTRODUCTION TO PROGRAMMING - PRACTICAL

Course Code	23U1PRBCA1
No. of credits	1
No. of contact hours	36

Course Description:

The course aims to introduce the field of programming and also gives an exposure to problem-solving through C and C++

Course Outcomes (COs)

CO1	Interpret the memory organization of variables, arrays, structures and union in C
CO2	Implement the functions and pointers using C and C++
CO3	Interpret dynamic memory allocation and file handling concepts in C and C++
CO4	Summarize Object Oriented Programming concepts
CO5	Demonstrate inheritance and generics concepts with examples

End Semester Examination Pattern: 75 Marks

The following guidelines should be followed regarding award of marks

(a) Lab Programs: 35 Marks

(i) Algorithm/pseudocode: 5 Marks

(ii)Implementing the work/Conducting the experiment: 15 Marks

(iii)Performance, result and inference (Usage of equipment and troubleshooting): 15 Marks

(b) Mini Project: 25 marks

(i) Implementation and demo: 15 Marks

(ii) Modification and results during the evaluation: 10 Marks

(c) Record: 5 Marks

(d) Viva Voice: 10 Marks

Continuous Internal Evaluation Pattern:

Attendance: 5 marks

Continuous Assessment: 20 Marks

Mini Project is evaluated using the rubrics given below.

	Unacceptable	Poor	Good	Excellent
Solution	An incomplete solution is implemented on the required platform. It does not compile and /or run	A complete solution is implemented on the required platform and uses the compiler specified. It runs, but has logical errors	A completed solution is tested and runs but does not meet all the specifications and / or work for all test data	A completed solution runs without errors. It meets all the specifications and works for all test data.
Program Design	Few of the selected structures are appropriate. Program elements are not well designed	Not all of the selected structures are appropriate. Some of the program elements are appropriately designed	The program design generally uses appropriate structures. Program elements exhibit good design	The program design uses appropriate structures. The overall program design is appropriate
Interface	User interaction is incomplete and does not meet specifications	User interaction minimally meets the specifications, but does not increase the usability	User interaction generally meets the specifications, and is acceptable to the user	User interaction is as specified and is natural

Part A: Programming in C and C++

1. Basic programming experiments to familiarization of data types and input output statements

2. Decision making, branching and looping statements

3. Function & Function calls

4. Array

5. Structure

6. Union

7. Bridging to C++- Pointers and functions with C and C++

8. Dynamic memory allocation

9. File handling in C & C++

10.Object oriented programming

11. Inheritance

12. Template Class and functions

Part B: Mini Project

Simple applications using C/ C++

SEMESTER I

DIGITAL SYSTEM DESIGN AND LINUX PRACTICAL

Course Code	23U1PRBCA2
No. of credits	1
No. of contact hours	36

Course Description:

The course aims to introduce the basic concepts of digital system design and familarize

Course Outcomes (COs)

CO1	Introducing and Implementing universal GATES
CO2	Implementing boolean expression using logic GATES
CO3	Implement file handling techniques
CO4	Implement and understand disk and network utilities

End Semester Examination Pattern: 75 Marks

The following guidelines should be followed regarding award of marks

(a) Digital System Design Lab Programs: 30 Marks

(i) Implementing the work/Conducting the experiment: 20 Marks

(ii) Performance, result and inference (Usage of equipment and troubleshooting): 10 Marks

(b) Linux Lab Programs: 30 marks

(i) Implementing the work/Conducting the experiment: 20 Marks

(ii)Performance, result and inference (Usage of equipment and troubleshooting): 10 Marks

(c) Record: 5 Marks

(d) Viva Voice: 10 Marks

Continuous Internal Evaluation Pattern:

Attendance: 5 marks

Continuous Assessment: 20 Marks

Part A: Digital System Design Lab using multisim.

List of Experiments

1. Realization of all Basic gates using Universal Gates (Both NAND and NOR) using multisim.

2. Simplification, Realization of Boolean Expressions by using Logic gates/Universal Gates using multisim

3. Realization of half/full adder and half/full subtractor using logic gates using multisim.

4. Design of a combinational circuit to convert from Binary to grey code converter and vice versa using multisim.

5. Realization of MUX / DEMUX using NAND gates using multisim.

6. To verify the truth table of one bit and two-bit comparators by using logic Gates using multisim.

7. Design of MOD N counter (synchronous/Asynchronous) using IC 7476 using multisim.

Part B: Linux Lab

List of programs

1. File Handling utilities
2. Text processing utilities
3. Network and processing utilities
4. Backup and restore utilities
5. Disk utilities
6. Filters
7. Shell Scripting

3.2 SEMESTER II

TEXT AND CONTEXT: A GUIDE TO EFFECTIVE READING AND WRITING

Course Code	23U2CCENG03
No. of credits	4
No. of contact hours	90

Course Description:

To develop the critical and analytical faculty of students and to improve their proficiency in reading, writing and presentation.

Course Outcomes (COs)

CO1	Understand the fundamental concepts of critical reasoning and to enable them to read and respond critically, drawing conclusions, generalizing, differentiating fact from opinion and creating their own arguments.
CO2	Assist the students in developing appropriate and impressive writing styles for various contexts.
CO3	Understand the structural imperfections and to edit what they have written.
CO4	Equip students for making academic presentations effectively and impressively.

Module 1: **(15 Hours)**

Critical Thinking

Introduction to critical thinking – Benefits - Barriers – Reasoning - Arguments - Deductive and inductive arguments – Fallacies - Inferential comprehension- Critical thinking in academic writing - Clarity - Accuracy – Precision - Relevance.

Module 2: **(20 Hours)**

Research for Academic Writing and the Writing Process

Data collection - Use of print, electronic sources and digital sources -Selecting key points - Note making, paraphrasing, summary – Documentation - Plagiarism – Title – Body paragraphs - Introduction and conclusion – Revising - Proof-reading

Module 3: (20 hours)

Accuracy in Academic Writing: Articles - Nouns and prepositions - Subject-verb agreement - Phrasal verbs - Modals - Tenses - Conditionals – Prefixes and suffixes – Prepositions - Adverbs – Relative pronouns - Passives - Conjunctions - Embedded questions - Punctuation – Abbreviations

Module 4: (20 hours)

Writing Models

Letters - Letters to the editor - Resume and covering letters - e-mail - Seminar papers - Project reports - Notices - Filling application forms - Minutes, agenda - Essays

Module 5: (15 hours)

Presentation Skills

Soft skills for academic presentations - Effective communication skills – Structuring the presentation - Choosing appropriate medium – Flip charts – OHP - PowerPoint presentation – Clarity and brevity - Interaction and persuasion - Interview skills –Group Discussions.

References:

1. Marilyn Anderson, Pramod K Nayar and Madhucchandra Sen. *Critical Thinking, Academic Writing and Presentation Skills*. Pearson Education and Mahatma Gandhi University.

SEMESTER II
DISCRETE MATHEMATICS

Course Code	23U2CPCMT2
No. of credits	4
No. of contact hours	72

Course Description:

The course aims to understand theory of computing, permutations, graph theory, solving system of equations and numerical methods for approximation.

Course Outcomes (COs)

CO1	Apply theory of computing, permutations in the area of communication networks, cryptography, network security
CO2	Apply graph theoretical algorithms to solve problems in communication networks.
CO3	Solving linear systems using the system of equations using Newton’s method, Gaussian elimination, Gauss-Jordan and Gauss- Seidel method
CO4	Interpret the different numerical methods for interpolation, differentiation, integration and solving set of ordinary differential equations.

Module 1: **(18 Hours)**

Combinatorics

The theory of counting. The multiplication rule, ordered sample and permutations, unordered samples without repetition, permutations involving indistinguishable objects, multinomial coefficient, unordered samples with repetition, permutation involving indistinguishable objects.

Module 2: **(18 Hours)**

Graphs and Algorithms

Leonhard Euler and the seven bridges of Konigsberg, trees and spanning trees, minimal spanning trees, binary trees and tree searching. Planar graphs and Euler's theorem, the shortest path problem, Dijkstra's Algorithm, two "all-pairs" Algorithm, Floyd's Algorithm and Warshall's Algorithm.

Module 3: **(18 Hours)**

Solution of equation

Fixed point iteration: x=g(x) method -Newton's method –Solution of linear system by Gaussian elimination and Gauss-Jordan method –Iterative method -Gauss- Seidel method - Inverse of a matrix by Gauss Jordan method.

Module 4: **(18 Hours)**

Numerical solutions for Calculus

Numerical Differentiation and Integration: Differentiation using interpolation formula – Numerical integration by trapezoidal and Simpson's 1/3 and 3/8 rules.

Ordinary differential equations: Single step methods: Taylor series method–Euler method for first order equation–Fourth order Runge–Kutta method for solving first and second order equations.

Text Books:

1. Engineering Mathematics, N.P. Bali, Manish Goyal
2. Petergray – Logic, Algebra and databases (chapter 3), Affiliated East West press pvt Ltd.
3. Robert J mcEliece, Robert B Ash and Carol Ash – Introduction to discrete mathematics (chapter 1,2 and 4), Mc.Graw Hill.

References:

1. James Ward Brown, Ruel V. Churchill, Complex Variables and Applications, 8th edition
2. S. Lipschutz, Set Theory and related topics, 2nd Edition., Schaum Outline Series, Tata McGraw-Hill Publishing Company, New Delhi.
3. R.G.Stoll, Set Theory and Logic, Dover Publications.
4. John Clark, Derek Allen Holton, A first look at graph theory, Allied Publishers.
5. Douglas B west, Introduction to Graph Theory, 2nd edition, Pearson Education.

SEMESTER II

OPERATING SYSTEM

Course Code	23U2CRBCA4
No. of credits	3
No. of contact hours	72

Course Description:

The aim of this course is to understand the concepts and principles of processes and processor management, concurrency and synchronization, memory management schemes, file system and secondary storage management, security and protection,

Course Outcomes (COs)

CO1	Identify mechanism to handle processes, memory, I/O devices, and files and develop an appropriate algorithm for it.
CO2	Discuss issues of Process Management including process structure, synchronization, scheduling and communication.
CO3	Interpret the reasons for deadlock state, and the solution methods to handle it
CO4	Differentiate type of memory management techniques used by Operating Systems
CO5	Appreciate the need of access control and protection in an operating system

Module 1: (18 Hours)

Introduction to Operating System

Introduction, Objectives and Functions of OS, Types of operating system, Evolution of OS, OS Structures, OS Components, OS Services, System calls, System programs

Processes: Process concept, Process scheduling, Co-operating processes, Operations on processes, Inter process communication.

Threads: Introduction to Threads, Single and Multi-threaded processes and its benefits, User and Kernel threads, Multithreading models, Threading issues.

Module 2: (18 Hours)

CPU Scheduling, Process Synchronization, Deadlock

CPU Scheduling: Basic concepts, Scheduling criteria, Scheduling Algorithms, Multiple

Processor Scheduling, Real-time Scheduling, Algorithm Evaluation, Process Scheduling Models.

Process Synchronization: Mutual Exclusion, Critical – section problem, Synchronization hardware, Semaphores, Classic problems of synchronization, Critical Regions, Monitors, OS Synchronization, Atomic Transactions.

Deadlocks: System Model, Deadlock characterization, Methods for handling Deadlocks, Deadlock prevention, Deadlock Avoidance, Deadlock Detection, Recovery from Deadlock**.**

Module 3: (18 hours)

Memory Management: Logical and physical Address Space, Swapping, Contiguous Memory Allocation, Paging, Segmentation with Paging. **Virtual Management**: Demand paging, Process creation, Page Replacement Algorithms- FIFO, Optimal page replacement algorithm, LRU, Allocation of Frames, Thrashing, Operating System Examples, Page size and other considerations, Demand segmentation

Module 4: (18 hours)

File and Disk Management

File-System Interface: File concept, Access Methods, Directory structure, File- system Mounting. **File-System Implementation**: File-System structure, File-System Implementations, Directory Implementation, Allocation Methods, Free-space Management, Efficiency and Performance, Recovery

Disk Management: Disk Structure, Disk Scheduling-FCFS, SSTF, C Scan, Look, C-Look Disk Management

Protection: Goals of Protection, Domain of Protection, Access Matrix, implementation of access matrix. **Security:** Security Problem, User Authentication, One – Time Password, Program Threats, System Threats, Cryptography- Features of cryptography -confidentiality, integrity, non-repudiation, authentication, Symmetric key and Asymmetric key cryptography

References:

1. Milan Milonkovic, Operating System Concepts and Design, 2nd Edition.
2. Tanenbaum, Operation System Concepts, 2nd Edition, Pearson Education
3. Silberschatz Galvin Gagne, Operating System, 6th Edition WSE WILEY Publication
4. William Stallings, Operating System, 4th Edition, Pearson Education.

SEMESTER II

PROBLEM SOLVING USING PYTHON

Course Code	23U2CRBCA5
No. of credits	3
No. of contact hours	72

Course Description:

The course aims at preparing the graduates for implementing PYTHON programs to solve computational problems in their professional life. The course content is decided to cover the essential programming fundamentals which can be taught within the given slots of the curriculum.

Course Outcomes (COs)

CO1	Create python programs by utilizing the data structures lists, tuple and dictionary.
CO2	Write programs using functions and packages
CO3	Identify the commonly used operations involving files and exceptions
CO4	Articulate the object-oriented programming concepts including encapsulation, inheritance and polymorphism as used in python
CO5	Apply the Python packages NumPy, Panda and visualization tools for data analysis

Module 1: **(15 Hours)**

Introduction to Python

Understanding Python-identifiers, variables, keywords, expressions and statements, evaluation of expressions, Operators and operands, operator precedence, indentation. Python Program Flow Control: Decision making- if, if else, elseif. Loops - for, while, for...else, while...else, Control statements using pass, continue, break. Strings and lists – string traversal, string slices and comparison with examples, The string module, character classification. List-List values, accessing elements, list membership, Lists and for loops, List operations, List slices, List deletion, Matrices. Tuples - mutability and tuples, tuple assignment, Tuples as return values, Tuple operations. Dictionaries – operations and methods.

Module 2: (15 Hours)

Python Functions, Modules and Packages

Function definition, calling functions, parameters and arguments, the return statement, type conversion and coercion, composition of functions, Lambda function, mathematical functions, user-defined functions, Recursion, Modules- Built-in modules, creating modules, import statements. Packages in Python -importing modules from a package.

Module 3: (12 hours)

Python Files and exceptions

Python file handling, open, write, read text files, writing variables, Directories in Python, Pickling, Exception Handling.

Module 4: (15 hours)

Python Object Oriented Programming

Introduction to classes and objects - class definition, attributes, instances, sameness, instances as arguments and return values. Constructor, class attributes and destructors, Inheritance.

Module 5: (15 hours)

Python for Data Analysis

Using Python for data analysis, essential python libraries, IPython, Jupyter notebook, NumPy basics, working with pandas. Case study on data science applications- Image analysis, Customer personality analysis.

Text Books

1. Allen Downey, Jeffrey Elkner, Chris Meyers, " How to think like a Computer Scientist-Learning with Python", Green Tea Press, First edition, 2020.
2. Mark Lutz,"Learning Python: Powerful Object-Oriented Programming" , O"Reilly Media Inc.,5th,2021.
3. Python for Data analysis: Data wrangling with Pandas, Numpy and IPython, Beijing:O'Reilly(1.2,1.3,2.2,4.1-4.5,5.1-5.3).

References:

1. Kenneth A. Lambert, B. L. Juneja, “Fundamentals of Python”, Cengage Learning India Pvt. Ltd., 2015.
2. S.A.Kulkarni, “Problem Solving and PYTHON Programming”, 2nd edition, Yes Dee Publishing Pvt Ltd, 2018
3. Mark Summerfield,”Programming in Python 3: A Complete Introduction to the Python Language”, Pearson Education, 2nd,2018
4. Yashavant Kanetkar ,Aditya Kanetkar ,”Let Us Python ”,BPB Publications, 1st Edition, 2019.
5. Allen Downey, “Learning with Python”, Dreamtec Press, 1st Edition, 2015
6. https://docs.python.org/3/reference/

SEMESTER II

DATA STRUCTURES AND ALGORITHMS USING C++

Course Code	23U2CRBCA6
No. of credits	3
No. of contact hours	72

Course Description:

This course covers the design, analysis, and implementation of basic data structures using C++. Algorithms operating on the data structures are also covered.

Course Outcomes (COs)

CO1	Illustrate the concepts of data structures, algorithms, asymptotic notation.
CO2	Apply algorithm for solving problems like sorting, searching, insertion and deletion of data
CO3	Implement the basic data structures arrays, stacks and queues using the programming language C++
CO4	Implement the operations creation, insertion, deletion, search and display on the data structure linked lists using the programming language C++
CO5	Discuss the applications of graph and trees to represent the flow of computation.

Module 1: **(18 Hours)**

Introduction to Data Structures:

Introduction to Data Structures- Classification of data structures: Primitive and non-primitive, elementary data organization

Notion of an Algorithm – Fundamentals of Algorithmic Problem Solving – Important Problem Types– Fundamentals of the Analysis of Algorithmic Efficiency –Asymptotic Notations and their properties.

Basic Search Techniques: Sequential search: Iterative and Recursive methods, Binary search: Iterative and Recursive methods, Comparison between sequential and binary search.

Sort: General background and definition, Bubble sort, Selection sort, Insertion sort, Merge sort, Quick sort.

Module 2: (18 Hours)

Stack and Queue

Templates in C++, Template Functions- Using Templates to Represent Container Classes, The Stack Abstract Data Type, The Queue Abstract Data Type

Stack – Definition, Array representation of stack, Operations on stack: Infix, prefix and postfix notations, Conversion of an arithmetic expression from Infix to postfix, Applications of stacks.

Queue: Definition, Array representation of queue, Types of queue: Simple queue, Circular queue, Double ended queue (deque), Priority queue, Operations on all types of Queues

Module 3: (18 hours)

Linked List

Single Linked List and Chains, Representing Chains in C++, Defining a Node in C++- Designing a Chain Class in C++- Pointer manipulation in C++- Chain Manipulation Operations, The Template Class Chain, Implementing Chains with Templates- Chain Iterators- Chain Operations- Reusing a Class. Definition, Components of linked list, Representation of linked list, Advantages and Disadvantages of linked list.

Types of linked list: Singly linked list, doubly linked list, Circular linked list, Operations on singly linked list: creation, insertion, deletion, search and display.

Module 4: (18 hours)

Tree Graphs and their Applications:

Definition: Tree terminology: Root, Node, Degree of a node and tree, Terminal nodes, Non-terminal nodes, Siblings, Level, Edge, Path, depth, Parent node, ancestors of a node.

Binary tree: Array representation of tree, Creation of binary tree. Traversal of Binary Tree: Preorder, Inorder and postorder, Binary tree, Complete binary tree, Binary search tree, Heap Tree. Graphs, Application of Graphs, Depth First search, Breadth First search

Text Books:

1. Data Structures using C++, Special Edition-MRCET, Tata McGraw-Hill Publishers.
2. Data structures, Algorithms and Applications in C++, S Sahni, University Press (India) Pvt.Ltd, 2nd edition, Universities Press Orient Longman Pvt. Ltd.

References:

1. Data structures and Algorithms in C++, Michael T.Goodrich, R.Tamassia and .Mount, Wiley student edition, John Wiley and Sons.
2. Data structures and Algorithm Analysis in C++, Mark Allen Weiss, Pearson Education. Ltd., Second Edition.
3. Data structures and algorithms in C++, 3rd Edition, Adam Drozdek, Thomson
4. Data structures using C and C++, Langsam, Augenstein and Tanenbaum, PHI.
5. Problem solving with C++, The OOP, Fourth edition, W.Savitch, Pearson education.

SEMESTER II

PROBLEM SOLVING USING PYTHON-PRACTICAL

Course Code	23U2PRBCA3
No. of credits	1
No. of contact hours	36

Course Description:

This laboratory course is meant for understanding and practicing the programming language Python.

Course Outcomes (COs)

CO1	Develop Python programs by making use of basic constructs- Decision controls, Looping controls, Lists, Tuple, Strings and Dictionaries
CO2	Design modular Python programs using functions and recursive functions
CO3	Design programs to create, access, modify, delete File
CO4	Develop data analysis programs using Numpy, Scipy, Pandas

End Semester Examination Pattern: 75 Marks

The following guidelines should be followed regarding award of marks

(a) Lab Programs: 35 Marks

(i) Algorithm/pseudocode: 5 Marks

(ii)Implementing the work/Conducting the experiment: 15 Marks

(iii)Performance, result and inference (Usage of equipment's and troubleshooting): 15 Marks

(b) Mini Project: 25 marks

(i) Implementation and demo: 15 Marks

(ii)Modification and results during the evaluation: 10 Marks

(c) Record: 5 Marks

(d) Viva Voce: 10 Marks

Continuous Internal Evaluation Pattern

Attendance: 5 marks

Continuous Assessment: 20 Marks

Mini Project is evaluated using the rubrics given below.

	Unacceptable	**Poor**	**Good**	**Excellent**
Solution	An incomplete solution is implemented on the required platform. It does not compile and /or run	A complete solution is implemented on the required platform and uses the compiler specified. It runs, but has logical errors	A completed solution is tested and runs but does not meet all the specifications and / or work for all test data	A completed solution runs without errors. It meets all the specifications and works for all test data.
Program Design	Few of the selected structures are appropriate. Program elements are not well designed	Not all of the selected structures are appropriate. Some of the program elements are appropriately designed	The program design generally uses appropriate structures. Program elements exhibit good design	The program design uses appropriate structures. The overall program design is appropriate
Interface	User interaction is incomplete and does not meet specifications	User interaction minimally meets the specifications, but does not increase the usability	User interaction generally meets the specifications, and is acceptable to the user	User interaction is as specified and is natural

Part A: Programming in Python

List of Programs

1. Basic programming experiments to familiarization of data types and input output statements

2. Decision making, branching and looping statements

3. Function & Function calls

    1. Function definitions and access

    2. Parameters and arguments

    3. Recursion

    4. Strings

    a) String traversal, join, slicing

b) String searching, Comparison

c) Other important String methods

5. Lists, Tuples and Dictionaries

a) Creation of List & List Operations

b) Tuple and Tuple operations

c) Creation of Dictionary and Operations

d) Comparison of List and Tuple

6. Matrix representation

a) Creating matrix

b) Matrix operations - addition, subtraction and multiplication

7. Files and Operations

a) Files - defining, opening/closing, read/write operations

b) Exceptions in Python

c) Pickling

8. Object Oriented Programming using Python

a) Creation of Classes & Instances, method calling

b) Constructor & Destructor concepts

c) Implementation of Inheritance

9. Python for Data Analysis

a) Operations on 1D, 2D, 3D Numpy arrays

b) Programmes using Scipy sub libraries

c) Data analysis using Pandas

Part B: Mini Project:

Simple Desktop/Web applications using python GUI toolkit

SEMESTER II

DATA STRUCTURES AND ALGORITHMS USING C++ - PRACTICAL

Course Code	23U2PRBCA4
No. of credits	1
No. of contact hours	36

Course Description:

This course covers the design, analysis, and implementation of basic data structures using C++. Algorithms operating on the data structures are also covered.

Course Outcomes (COs)

CO1	Understand the concept of data types, algorithms, Asymptotic notation.
CO2	Apply Algorithm for solving problems like sorting, searching, insertion and deletion of data
CO3	Understand basic data structures such as arrays, linked lists, stacks and queues.
CO4	Solve problem involving graphs, trees

End Semester Examination Pattern: 75 Marks

The following guidelines should be followed regarding award of marks

(a) Lab Programs: 55 Marks

(i) Algorithm/pseudocode:10 Marks

(ii)Implementing the work/Conducting the experiment: 35 Marks

(iii)Performance, result and inference (Usage of equipment and troubleshooting): 10 Marks

(b) Record: 10 Marks

(c) Viva Voce: 10 Marks

Continuous Internal Evaluation Pattern:

Attendance: 5 marks

Continuous Assessment: 20 Marks

Part A: Data Structures and algorithm using C++

List of Programs

1. Basic Search Techniques:

   i. Sequential search: Iterative and Recursive methods

   ii. Binary search: Iterative and Recursive methods

2. Sort:

   i. Bubble sort

   ii. Selection sort

   iii. Insertion sort

   iv. Merge sort

   v. Quick sort

3. Stack

   i. Infix, prefix and postfix notations

   ii. Conversion of an arithmetic expression from Infix to postfix

4. Queue

   i. Simple queue – insert, update, delete, search

   ii. Circular queue- – insert, update, delete, search

   iii. Double ended queue (deque) -– insert, update, delete, search

   iv. Priority queue- – insert, update, delete, search

5. Linked List

   i. Operations on singly linked list: creation, insertion, deletion, search and display.

6. Tree

   i. Traversal of Binary Tree: Preorder, Inorder and postorder

7. Graph

   ii. Depth First search, Breadth First search

3.3 SEMESTER III

BASIC STATISTICS

Course Code	23U3CRBCA7
No. of credits	4
No. of contact hours	72

Course Description:

This course covers the basics of statistics ie; measure of central tendency and dispersion, probability, correlation, regression and index numbers.

Course Outcomes (COs)

CO1	Understand the concept of central tendency and basics of statistics
CO2	Understand the concept of dispersion and implement the concepts of correlation and regression.
CO3	Implement the concept of basic probability and statistics
CO4	Implement the concept of Index numbers

Module 1 **(15 Hours)**

Measures of Central Tendency – Mean; Median; Mode; Geometric Mean; Harmonic Mean and Properties, Combined mean, Partition values- quartiles, percentiles, deciles.

Module 2 **(20 Hours)**

Absolute and Relative measures of Dispersion – Range, Quartile Deviation, Percentiles, Deciles, Box Plot, Mean Deviation, Standard Deviation, Coefficient of Variation.

Correlation and Regression – Basic Concepts, Pearson Correlation Coefficient, Rank Correlation Coefficient, Regression equations (without derivation)

Module 3 **(20 Hours)**

Idea of Permutations and Combinations, Probability Concepts – Random Experiment, Sample Space, Events, Probability Measure, Approaches to Probability – Classical, Statistical and Axiomatic, Addition Theorem (upto 3 evens) Conditional Probability, Independence of events, Multiplication theorem (upto 3 events), Total Probability Law, Baye’s Theorem and its applications

Module 4 **(17 Hours)**

Index Numbers – definition, Simple Index Numbers; Weighted Index Numbers – Laspeyer’s Paasche’s and Fisher’s Index Numbers, Test of Index Numbers, Construction of Index Numbers, Cost of Living Index Numbers – Family Budget Method, Aggregate Expenditure Method.

Time Series – Components of time series and measures of time series Analysis.

Text Books:

1. S.P. Gupta (1987), Statistical Methods, Sultan Chand & Sons Delhi
2. S.C. Gupta and V.K. Kapoor, Fundamentals of Mathematical Statistics, Sultan Chand and Sons.

References:

1. Parimal Mukhopadhya, Mathematical Statistics, New Central Book Agency (p) Ltd, Calcutta
2. Murthy M.N, Sampling theory and Methods, Statistical Publishing Society Calcutta. Agarwal B L., Basic Statistics

SEMESTER III

SOFTWARE ENGINEERING

Course Code	23U3CRBCA8
No. of credits	3
No. of contact hours	72

Course Description:

This course covers the fundamentals of software engineering, including understanding system requirements, finding appropriate engineering compromises, effective methods of design, coding, and testing, team software development, and the application of engineering tools.

Course Outcomes (COs)

CO1	Design the software engineering project models including research and a feasibility study prior to embarking on a development project.
CO2	Construct UML diagrams for representing the features and operation in the software projects
CO3	Discuss the design patterns appropriate for software development and the concepts of agile methodology and SCRUM framework
CO4	Generate test suites for testing the software projects
CO5	Plan and schedule the software project development by analysing the cost, risks involved

Module 1: **(15 Hours)**

Introduction to Software Engineering:

Characteristics of Software. Software Engineering models: Predictive software engineering models, model approaches, predictive and adaptive waterfall, waterfall with feedback (Sashimi), incremental waterfall, V model; Prototyping and prototyping models.

Software requirements specification, Eliciting Software requirements, Requirements modeling, Requirements documentation. Use cases and User stories.

Cohesion and Coupling Function oriented design: Overview of SA/SD Methodology, Developing the DFD model of a system, Structured Design, User Interface design: Characteristics of a good user interface, Types of user interfaces

Module 2: **(15 Hours)**

Object Oriented Modeling and Design using UML

Object state and properties, Behavior, Methods, Messages, Object Oriented system development life cycle. Benefits of OOMethodology Factional view(models):, Use case diagram ,Activity diagram , Static structural view (Models) Behavioral (Dynamic structural view):,State diagram, Interaction diagrams: Approaches for developing dynamic systems:, Architectural view: Reuse: Libraries, Frame works components and Patterns.

Module 3: **(15 Hours)**

Design Patterns:

Basic concepts of Design patterns, How to select a design pattern, Creational patterns, Structural patterns, Behavioural patterns. Concept of Anti-patterns. Concepts of Agile Development methodology; Scrum Framework.

Module 4: **(15 Hours)**

Software Testing:

Software testing principles, Program inspections and walkthroughs, Program reviews; Unit Testing frameworks, The xUnit Architecture. Assertions, Custom Assertions, single condition tests, testing for expected errors, Abstract test. Blackbox testing: Equivalence class testing, Boundary value testing, Decision table testing, Pairwise testing, State transition testing, Use-case testing; White box testing: control flow testing, Data flow testing. Testing automation: Defect life cycle; Regression testing, Testing automation; Testing non-functional requirements.

Module 5: **(12 Hours)**

Software Project Management

Measures And Measurements – ZIPF’s Law – Software Cost Estimation – Function Point Models – COCOMO Model – Delphi Method – Scheduling – Earned Value Analysis – Error Tracking – Software Configuration Management – Program Evolution Dynamics – Software Maintenance – Project Planning – Project Scheduling– Risk Management – CASE Tools

Text Books:

1. Philip A. Laplante, What Every Engineer Should Know about Software Engineering, CRC Press

2. Murali Chemuturi, Mastering Software Quality Assurance: Best Practices, Tools and Technique for Software Developers, J Ross Publishing

3. Erich Gamma et. al., Design Patterns: Elements of Reusable Object-Oriented Software, AddisonWesley

4. Alistair Cockburn and Robert Cecil Martin, Agile Software Development: The Cooperative Game (2ndedition), Addition Wesley

5. Ken Schwaber, Agile Software Development with Scrum, Pearson

6. Lisa Crispin, Agile Testing: A Practical Guide for Testers and Agile Teams, Adison Wesley

7. Glenford J. Myers, et. al., The Art of Software Testing, Wiley

8. Lee Copeland, A Practitioner's Guide to Software Test Design, Artech House Publishers

9. Jez Humble and David Farley, Continuous Delivery: Reliable Software Releases through Build, Test, and Deployment Automation, Pearson Education

10. Roger Pressman, Software Engineering: A Practitioner‟s Approach, 7th Edition, McGraw-Hill, 2019.

11. Ian Sommerville, Software Engineering, 9th Edition, Addision Wesley.

12. Pankaj Jalote, A Concise Introduction to Software Engineering, Springer,2008.

13. William E. Lewis, Software Testing and Continuous Quality Improvement, Third Edition,Auerbach Publications, 2008

SEMESTER III

RDBMS

Course Code	23U3CRBCA9
No. of credits	4
No. of contact hours	72

Course Description:

The course is intended to provide a strong formal foundation in database concepts, technology and practice to groom students into well-informed database application developers.

Course Outcomes (COs)

CO1	Construct the data models using the entity-relationship and developing database designs.
CO2	Explain the Relational Algebra and relational calculus concepts and use it to translate sql queries to Relational Algebra statements and vice versa.
CO3	Apply the Structured Query Language (SQL) syntax to develop relational model
CO4	Apply normalization techniques to reduce the anomalies associated with the database
CO5	Interpret the needs of database processing and the techniques to control the consequences of concurrent data access.
CO6	Create a simple database system and demonstrate competence with the fundamental tasks involved with modelling, designing, and implementing a DBMS

Module 1: **(15 Hours)**

Introduction: Purpose of Database System -– Views of data – Data Models – Database Languages –– Database System Architecture – Database users and Administrator – Entity Relationship model (E-R model) – E-R Diagrams

Module 3: **(15Hours)**

The relational Model: Integrity Constraints, creation of relations from E-R model.

Relational Algebra and operations: Select, Project, Union, Set different, Cartesian product, Rename

Relational Calculus: Domain Relational Calculus – Tuple Relational Calculus.

Module 3: (15 Hours)

SQL fundamentals

Data types, Data Constraints, Column level & table Level Constraints, working with Tables. Basic SQL Commands: Defining constraints on the table, Defining Integrity Constraints, ALTER TABLE Command, Select Command, Logical Operator, Range Searching, Pattern Matching, Grouping data from Tables in SQL, Manipulation Data in SQL.

Joining Multiple Tables (Equi Joins), Joining a Table to itself (self Joins), Sub queries Union, intersect & Minus Clause, Creating view, Renaming the Column of a view, Granting Permissions, - Updating, Selection, Destroying view Creating Indexes, Creating and managing User.Views, Procedure and Triggers

Introduction to Distributed Databases and Client/Server Databases, NoSQL

Module 4: (12 Hours)

Database Design: Functional Dependencies – Non-loss Decomposition – Functional Dependencies – First, Second, Third Normal Forms, Dependency Preservation – Boyce/Codd Normal Form-Multi-valued Dependencies and Fourth Normal Form – Join Dependencies and Fifth Normal Form

Module 5: (15 Hours)

Transactions: Transaction Concepts - Transaction Recovery – ACID Properties – System Recovery – Media Recovery – Two Phase Commit - Save Points – SQL Facilities for recovery –Concurrency – Need for Concurrency – Locking Protocols – Two Phase Locking – Intent Locking – Deadlock- Serializability – Recovery Isolation Levels – SQL Facilities for Concurrency.

References:

1. Abraham Silberschatz, Henry F. Korth, S. Sudharshan, Database System Concepts 5th Edition, Tata McGraw Hill
2. Ramez Elmasri, Shamkant B. Navathe, Fundamentals of Database System 4th Edition, Pearson Addision Wesley
3. Raghu Ramakrishnan, Database Management Systems 3rd Edition, McGraw Hill

SEMESTER III

COMPUTER NETWORKS

Course Code	23U3CRBCA10
No. of credits	4
No. of contact hours	90

Course Description:

To uncover and understand the current directions of computer networks

Course Outcomes (COs)

CO1	Familiarize the basic components in a computer network, network topologies and the types of networks.
CO2	Differentiate between the network communication reference models OSI and TCP/IP
CO3	Demonstrate the data transmission methodologies FDM, TDM, WDM, switching techniques and the internet protocols IPV4 and IPV6.
CO4	Explain the functions of data link and transportation layer
CO5	Explain the types of wireless communication, IEEE standards used, network utility commands
CO5	Summarize the network security threats associated in a network and the policies including DES, AES, RSA.

Module 1: **(15 Hours)**

Networking Fundamentals and Network Devices

Introduction to Communication: Basics of Network & Networking; Advantages of Networking; **Types of Networks**: LAN;MAN;WAN; **Network Terms**: Host; Workstations; Server; Client; Node; Types of Network Architecture: Peer-to-Peer & Client/Server; Workgroup Vs. Domain; **Network Devices**- NIC- functions of NIC; Hub; Switch; Bridge; Router; Gateways; And Other Networking Devices; and modem;

Network Topologies: Types of Topologies; Logical and physical topologies; selecting the Right Topology; **Types of Transmission Media**: Wired and wireless; **Types of Cables:** Wiring Standards and Cabling- straight through cable; crossover cable; rollover cable; media connectors (Fiber optic; Coaxial; and TP etc.) **Models of Network** Introduction of OSI model; Seven layers of OSI model; Functions of the seven layers; Introduction of TCP/IP Model; Comparison between OSI model & TCP/IP model; Overview of Ethernet Addresses;

Module 2: **(20 Hours)**

Digital representation of information: properties of signals; **Transmission modes**: parallel and serial transmission; properties of digital transmission systems: Bandwidth, Signal-to- noise ratio, data rate Multiplexing: FDM; TDM; WDM; SONET;

Switching techniques: Packet switching; structure of packet switching; circuit switching. **Types of network Protocols:** Internet Protocol (IP); IP standards; versions; functions; IPv4 addressing; IPv4 address Classes; IPv4 address types; Subnet Mask; Default Gateway; Public & Private IP Address; methods of assigning IP address; IPv6 address; types; assignment; Data encapsulation; The IPv4 Datagram Format; The IPv6 Datagram Format; Internet Control Message Protocol (ICMP); ICMPv4; ICMPv6; Internet Group Management Protocol(IGMP);

Module 3: **(20 Hours)**

Transport Layer: Transmission Control Protocol (TCP); User Datagram Protocol (UDP); Overview of Ports & Sockets; **Application Layer:** DHCP; DNS; HTTP/HTTPS; FTP; TFTP; SFTP; Telnet; Email: SMTP; POP3/IMAP. **Data Link Layer:** Ethernet; Ethernet standards; Ethernet Components; Point-to-Point Protocol (PPP);PPP standards; Address Resolution Protocol; Message format;

Framing: fixed size framing; variable size framing **Routing algorithms:** Shortest path (Bellman Ford Algorithm) Dijkstras' algorithm. **Flow Control:** Noiseless Channel Protocol: simplest protocol; stop and wait protocol. Noisy channel protocol: Stop and wait ARQ; Goback N ARQ; selective repeat ARQ; piggy backing. **Error detection and correction:** Types of errors; Redundancy- detection and correction. Parity check; polynomial codes Hamming distance – minimum Hamming distance. **Congestion control algorithms:** Leaky bucket algorithm; Token bucket algorithm. Quality of services (QOS) – Integrated Services

Module 4: (20 Hours)

WAN Technology

Wireless Networking: Wireless Technology; Benefits of Wireless Technology; **Types of Wireless Networks:** Ad-hoc mode; Infrastructure mode; **Wireless network Components:** Wireless Access Points; Wireless NICs; **Wireless LAN standards:** IEEE 802.11a; IEEE 802.11b; IEEE 802.11g; wireless LAN modulation techniques; **wireless security Protocols:** WEP;WPA; 802.1X; **Network Utilities commands:** ping; traceroute; tracert; ipconfig; arp; nslookup; netstat; nbtstat; Hardware trouble shooting tools; system monitoring tools

Module 5: (15 Hours)

Network security: Authentication and Authorization; Public key encryption; Private key Encryption symmetric-key algorithms – DES, Triple DES, AES RSA; Tunneling and Encryption Protocols: IPSec; SSL and TLS; Firewall; Other Security Appliances; Security Threats. **Network Operating Systems**: Microsoft Operating Systems; Novell NetWare; UNIX and Linux Operating Systems; Macintosh Networking;

Text Books:

1. Andrew s. Tanenbaum, David j. Wetherall, Computer Networks, sixth edition, Pearson Publications 2021.
2. CCNA, Cisco Certified Network Associate: Study Guide (With CD), 5th Edition
3. CCENT/CCNA ICND1, Official Cert Guide, 3rd Edition (Paperback)

References:

1. CCNA, Routing Protocols and Concepts CCNA Exploration Companion Guide (With CD) (Paperback), Pearson
2. CCNA, Exploration Course Booklet Routing Protocols and Concepts Version 4.0 (Paperback), Pearson

SEMESTER III

PROGRAMMING IN JAVA

Course Code	23U3CRBCA11
No. of credits	3
No. of contact hours	72

Course Description:

To understand the fundamentals of programming in java.

Course Outcomes (COs)

CO1	Explain the basic concepts of Java Programming
CO2	Develop object-oriented programming in Java, including defining classes, invoking methods, using libraries.
CO3	Demonstrate the designing, implementing, testing and debugging graphical user interfaces in Java.
CO4	Illustrate Java Swings for designing GUI applications

Module 1: **(15 Hours)**

History, Overview of Java, Object Oriented Programming, A simple Programme, Two control statements - if statement, for loop, using Blocks of codes, Lexical issues - White space, identifiers, Literals, comments, separators, Java Key words. Data types: Integers, Floating point, characters, Boolean, A closer look at Literals, Variables, Type conversion and casting, Automatic type promotion in Expressions Arrays.

Operators: Arithmetic operators, The Bit wise operators, Relational Operators, Boolean Logical operators, Assignment Operator, Operator Precedence. Control Statements: Selection Statements - if, Switch: Iteration Statements - While, Do-while, for Nested loops, Jump statements.

Module 2: (15 Hours)

Classes

Class Fundamentals, Declaring objects, Assigning object reference variables, Methods, constructors, “this” keyword, finalize ( ) method A stack class, Over loading methods, using objects as parameters, Argument passing, Returning objects, Recursion, Access control, Introducing final, understanding static, Introducing Nested and Inner classes, Using command line arguments.

Inheritance: Inheritance basics, Using super, method overriding, Dynamic method Dispatch, using abstract classes, using final with Inheritance.

Module 3: (15 Hours)

Packages

Definition, Access protection importing packages, Interfaces: Definition implementing interfaces.

Exception Handling: Fundamental, Exception types, Using try and catch, Multiple catch clauses, Nested try Statements, throw, throws, finally, Java’s Built - in exception, using Exceptions.

Module 4: (15 Hours)

Multithreaded Programming

The Java thread model, The main thread, Creating a thread, Creating multiple thread, Creating a thread, Creating multiple threads, Using isalive() and Join(), Thread - Priorities, Synchronization, Inter thread communication, suspending, resuming and stopping threads, using multithreading. 1/0 basics, reading control input, writing control output, Reading and Writing files, Applet Fundamentals, the AWT package, AWT Event handling concepts The transient and volatile modifiers, using instance of using assert.

Module 5: (12 Hours)

JAVA Database Connectivity (JDBC)

Database connectivity: JDBC architecture, JDBC Drivers, the JDBC API: loading a driver, connecting to a database, Creating and executing JDBC statements, Handling SQL exceptions, Accessing result sets: Types of result sets, Methods of result set interface. An example JDBC application to query a database.

References:

1. Herbert. The complete referenc 1. Yashavant Kanetkar - Let Us Java – 3dr Edition – BPB - 2017
2. Cay S. Horstmann - Core Java, Volume I: Fundamentals – 12th Edition – Oracle - 2021
3. Lee Stemkoski, James Cona - Developing Graphics Frameworks with Java and OpenGL - CRC Press – 2022
4. web Daniel Leuck, Marc Loy, and Patrick Niemeyer - Learning Java: An Introduction to Real-World Programming with Java - 2021e Java –2, 5th Edition, Schildt Pub. TMH.
5. Rogers Cedenhead and Leura Lemay, SAMS teach yourself Java – 2, 3rd Edition, Pearson Education.

SEMESTER III

RDBMS - PRACTICAL

Course Code	23U3PRBCA5
No. of credits	1
No. of contact hours	36

Course Description:

The course is intended to provide a strong formal foundation in database concepts, technology and practice to groom students into well-informed database application developers.

Course Outcomes (COs)

CO1	Construct the data models using the entity-relationship and developing database designs.
CO2	Apply the Structured Query Language (SQL) syntax to develop relational model
CO3	Apply normalization techniques to reduce the anomalies associated with the database
CO4	Interpret the needs of database processing and the techniques to control the consequences of concurrent data access.
CO5	Create a simple database system and demonstrate competence with the fundamental tasks involved with modelling, designing, and implementing a DBMS

End Semester Examination Pattern: 75 Marks

The following guidelines should be followed regarding award of marks

(a) Lab Programs: 35 Marks

(i) Algorithm/pseudocode: 5 Marks

(ii)Implementing the work/Conducting the experiment: 15 Marks

(iii)Performance, result and inference (Usage of equipments and troubleshooting): 15 Marks

(b) Mini Project: 25 marks

(i) Implementation and demo: 15 Marks

(ii)Modification and results during the evaluation: 10 Marks

(c) Record: 5 Marks

(d) Viva Voce: 10 Marks

Continuous Internal Evaluation Pattern:

Attendance: 5 marks

Continuous Assessment: 20 Marks

Mini project is evaluated using the rubrics given below.

	Unacceptable	**Poor**	**Good**	**Excellent**
Solution	An incomplete solution is implemented on the required platform. It does not compile and /or run	A complete solution is implemented on the required platform and uses the compiler specified. It runs, but has logical errors	A completed solution is tested and runs but does not meet all the specifications and / or work for all test data	A completed solution runs without errors. It meets all the specifications and works for all test data.
Program Design	Few of the selected structures are appropriate. Program elements are not well designed	Not all of the selected structures are appropriate. Some of the program elements are appropriately designed	The program design generally uses appropriate structures. Program elements exhibit good design	The program design uses appropriate structures. The overall program design is appropriate
Interface	User interaction is incomplete and does not meet specifications	User interaction minimally meets the specifications, but does not increase the usability	User interaction generally meets the specifications, and is acceptable to the user	User interaction is as specified and is natural

List of Experiments

Part A: RDBMS

1. SQL Commands
   a. Data Definition Language commands,

b. Data Manipulation Language commands,
c. Data Control Language commands and
d. Transaction Control Language commands

2. Select Statements with all clauses/options
3. Nested Queries
4. Join Queries
5. Views
6. High level programming language extensions (Control structures, Procedures and Functions)

Part B: Mini Project Database Design and implementation

SEMESTER III

PROGRAMMING IN JAVA-PRACTICAL

Course Code	23U3PRBCA6
No. of credits	1
No. of contact hours	36

Course Description:

To understand the fundamentals of programming in java.

Course Outcomes (COs)

CO1	Explain basic concepts of Java Programming
CO2	Develop object-oriented programming in Java, including defining classes, invoking methods, using libraries.
CO3	Demonstrate the designing, implementing, testing and debugging graphical user interfaces in Java.
CO4	Illustrate Java Swings for designing GUI applications

End Semester Examination Pattern: 75 Marks

The following guidelines should be followed regarding award of marks

(a) Lab Programs: 35 Marks

(i) Algorithm/pseudocode: 5 Marks

(ii)Implementing the work/Conducting the experiment: 15 Marks

(iii)Performance, result and inference (Usage of equipments and troubleshooting): 15 Marks

(b) Mini Project: 25 marks

(i) Implementation and demo: 15 Marks

(ii)Modification and results during the evaluation: 10 Marks

(c) Record: 5 Marks

(d) Viva Voce: 10 Marks

Continuous Internal Evaluation Pattern:

Attendance: 5 marks

Continuous Assessment: 20 Marks

Mini Project is evaluated using the rubrics given below.

	Unacceptable	**Poor**	**Good**	**Excellent**
Solution	An incomplete solution is implemented on the required platform. It does not compile and /or run	A complete solution is implemented on the required platform and uses the compiler specified. It runs, but has logical errors	A completed solution is tested and runs but does not meet all the specifications and / or work for all test data	A completed solution runs without errors. It meets all the specifications and works for all test data.
Program Design	Few of the selected structures are appropriate. Program elements are not well designed	Not all of the selected structures are appropriate. Some of the program elements are appropriately designed	The program design generally uses appropriate structures. Program elements exhibit good design	The program design uses appropriate structures. The overall program design is appropriate
Interface	User interaction is incomplete and does not meet specifications	User interaction minimally meets the specifications, but does not increase the usability	User interaction generally meets the specifications, and is acceptable to the user	User interaction is as specified and is natural

List of Programs

Part A

1. Write a program to check whether two strings are equal or not.
2. Write a program to display reverse string.
3. Write a program to find the sum of digits of a given number.
4. Write a program to display a multiplication table.
5. Write a program to display all prime numbers between 1 to 1000.
6. Write a program to insert element in existing array.
7. Write a program to sort existing array.
8. Write a program to create object for Tree Set and Stack and use all methods.

9. Write a program to check all math class functions.
10. Write a program to execute any Windows 95 application (Like notepad, calculator etc)
11. Write a program to find out total memory, free memory and free memory after executing garbage Collector (gc).

Part B

12. Write a program to copy a file to another file using Java to package classes. Get the file names at run time and if the target file is existed then ask confirmation to overwrite and take necessary actions.
13. Write a program to get file name at runtime and display number f lines and words in that file.
14. Write a program to list files in the current working directory depending upon a given pattern.
15. Create a textfield that allows only numeric value and in specified length.
16. Create a Frame with 2 labels, at runtime display x and y command-ordinate of mouse pointer in the labels.

3.4 SEMESTER IV

WEB TECHNOLOGIES

Course Code	23U4CRBCA12
No. of credits	4
No. of contact hours	72

Course Description:

Understand the concepts of HTML, CSS, JavaScript, PHP-MySQL, Python flask to create web-based applications

Course Outcomes (COs)

CO1	Learn HTML elements and gain to explore CSS
CO2	Gain knowledge on writing JavaScript program
CO3	Learn programming knowledge in server-side scripting language PHP
CO4	Understand PHP-MySQL, its queries and the ability to establish the connection
CO5	Understand the concepts of python programming concepts and able to write a web application using python flask

Module 1: (16 Hours)

Introduction to Internet: Concept of WWW, Internet and WWW, HTTP Protocol: Request and Response, Web browser and Web servers

HTML: Basics of HTML, formatting and fonts, commenting code, color, hyperlink, lists, tables, images, forms, XHTML, frames and frame sets, Overview and features of HTML5.

Style sheets: Need for CSS, introduction to CSS, basic syntax and structure, using CSS, background images, colors and properties, XML, Introduction to Bootstrap

Module 2: (14 Hours)

General Syntactic Characteristics- Primitives, Operations, and expressions, Screen Output and Keyboard Input, Control Statements, Object Creation and Modification, Arrays, Functions, Callback Functions, Java Script HTML DOM, AJAX - Introduction, advantages &

disadvantages, AJAX based web application, jQuery -Introduction to jQuery: Overview and Basics.

Module 3: **(16 Hours)**

Introduction to PHP: Basics, Variables, Operators, Conditional statements and Switch case, Flow Control and Loops, Array, For each Loop, Global Array, String inbuilt functions, Math functions, Array Inbuilt functions.

Module 4: **(12 Hours)**

Using HTML Forms: PHP form handling, get data sent from form fields through GET and POST method, sessions and cookies.

Introduction to PHP MySQL: PHP-MySQL Connection overview, 3 different approaches – procedure, object oriented, PDO; PHP-MySQL function to connect to database, access database, fetch result.

Module 5: **(14 Hours)**

Introduction to Flask: Flask overview, Routing, Static files, Rendering template, Request and response, Mini project-web application with Flask.

Text Books:

1. Patrick Carey- New Perspectives HTML5 and CSS3: Comprehensive - 2017 -7 th Edition
2. Laurence Lars Svekis, Maaike van Putten, Rob Percival - JavaScript from Beginner to Professional: Learn JavaScript quickly by building fun, interactive, and dynamic web apps, games, and pages –2021 - Kindle Edition
3. Steven Holzner-2008-PHP: The Complete Reference, McGraw-Hill Higher Education

SEMESTER IV

INTRODUCTION TO CLOUD TECHNOLOGY

Course Code	23U4VCBCA1
No. of credits	4
No. of contact hours	72

Course Description:

To impart fundamentals concepts and applications in the area of cloud computing.

Course Outcomes (COs)

CO1	Understand the basics of cloud computing.
CO2	Know the Key concepts of Cloud Infrastructure Mechanisms.
CO3	Understand different Cloud Computing Architecture.
CO4	Understand the fundamentals of Cloud security.
CO5	Understand about cloud providers and case studies.

Module 1: (15 Hours)

Introduction

Introduction to Cloud Computing, History and Evolution of Cloud Computing, Types of clouds, Private Public and hybrid clouds, Cloud Computing architecture, Cloud computing infrastructure, Merits of Cloud computing, Cloud computing delivery models and services (IaaS, PaaS, SaaS), Obstacles for cloud technology, Cloud vulnerabilities, Cloud challenges, Practical applications of cloud computing.

Module 2: (15 Hours)

Cloud Infrastructure Mechanisms

Logical Network Perimeter, Virtual Server, Cloud Storage Device, Cloud Usage Monitor, Resource Replication. Specialized Cloud Mechanisms: Automated Scaling Listener, Load Balancer, SLA Monitor, Pay- per-Use Monitor, Audit Monitor, Failover System, Hypervisor, Resource Cluster, Multi-Device Broker- State Management Database

Module 3: (15 Hours)

Different Cloud Architecture

Cloud Architectures: Workload Distribution Architecture, Resource Pooling Architecture, Dynamic Scalability Architecture, Elastic Resource Capacity Architecture, Service Load Balancing Architecture, Cloud Bursting Architecture, Elastic Disk Provisioning Architecture, Redundant Storage Architecture , Hypervisor Clustering Architecture, Load Balanced Virtual Server Instances Architecture

Non-Disruptive Service Relocation Architecture, Zero Downtime Architecture, Cloud Balancing Architecture, Resource Reservation Architecture, Dynamic Failure Detection and Recovery Architecture, Bare-Metal Provisioning Architecture, Rapid Provisioning Architecture, Storage Workload Management Architecture.

Module 4: (15 hours)

Fundamentals of Cloud Security

API Logging and Auditing, Fundamental Cloud Security: Basic Terms and Concepts: Threat Agents, Cloud Security Threats, Additional Considerations.

Module 5: (12 hours)

Cloud Providers, Study of OpenStack and AWS clouds. Case Studies of Cloud Management Software.

Text Books:

1. Thomas Erl, Zaigham Mahmood, and Ricardo Puttini, Cloud Computing: Concepts, Technology & Architecture, Prentice Hall, ISBN-13: 978-0-13-338752
2. John W. Rittinghouse and James F. Ransome, Cloud Computing: Implementation, Management, and Security, CRC Press
3. Toby Velte, Anthony Velte, Robert Elsenpeter, Cloud Computing, A Practical App roach,TMH
4. Kumar Saurabh, Cloud Computing insights into New Era Infrastructure, Wiley India

References:

1. George Reese, Cloud Application Architectures: Building Applications and Infrastructure in the Cloud, O'Reilly
2. Rajkumar Buyya, James Broberg, Andrzej M. Goscinski, John Wiley, Cloud Computing: Principles and Paradigms, Sons Publications

SEMESTER IV

FUNDAMENTALS OF DATA CENTRE AND STORAGE

Course Code	23U4VCBCA2
No. of credits	4
No. of contact hours	72

Course Description:

This course contains the basics of data center design and storage.

Course Outcomes (COs)

CO1	Explain the fundamental concepts of information storage.
CO2	Discuss about Backup and recovery
CO3	Summarize the concept of local replication
CO4	Illustrate the basics concept of data centers and cloud data center topologies
CO5	Apply the concepts of server virtualization and networking

Module 1: (15 Hours)

Introduction to Information storage: Core elements, Components of a Storage System Environment – Host –Connectivity – Storage, Disk Drive Components –Platter – Spindle - Read/Write Head - Actuator Arm Assembly - Controller - Physical Disk Structure

Module 2: (15 Hours)

Backup and Recovery

Backup Purpose -Disaster Recovery - Operational Backup –Archival, Backup Considerations, Backup Granularity, Recovery Considerations, Backup Methods, Backup Process, Backup and Restore Operations, Backup Topologies - Serverless Backup, Backup Technologies - Backup to Tape - Physical Tape Library - Backup to Disk - Virtual Tape Library

Module 3: (15 Hours)

Local Replication

Source and Target -Uses of Local Replicas, Data Consistency - Consistency of a Replicated File System - Consistency of a Replicated Database, Local Replication Technologies - Host-Based Local Replication - Storage Array-Based Replication, Res tore and Restart Considerations - Tracking Changes to Source and Target, Creating Multiple Replicas, Management Interface

Module 4: (15 Hours)

Introduction to Data Centers

Data Centers Defined, Data Center Goals, Data Center Facilities, Roles of Data Centers in the Enterprise, Roles of Data Centers in the Service Provider Environment, Application Architecture Models. The Client/Server Model and Its Evolution, The n-Tier Model, Multitier Architecture Application Environment, Data Center Architecture.

Data Center Storage types: DAS, NAS, SAN

Cloud Data Center Networking Topologies: Traditional Multi-tiered Enterprise Networks· Data Center Network Switch Types, Flat Data Center Networks, Rack Scale Architectures, Network Function Virtualization

Module 5: (12 Hours)

Server Virtualization and Networking

Server Virtualization and Networking: VM Overview**-** type of VMs and Hypervisors**,** Virtual Switching. PCI Express Edge Virtual Bridging. VM Migration Network Virtualization: Multi-tenant Environments· Traditional network Tunneling Protocols· VXLAN· NVGRE· Tunnel Locations· Load Balancing.

References:

1. Gary Lee (2014), Cloud Networking Understanding Cloud based Data Center Networks, Elsevier
2. Gary Oreinstein (2006), IP Storage Networking, Addison Wesley Professional
3. G. Somasundaram, Alok Srivastava (2009), Information Storage and Management 1st edition, Wiley
4. Kailash Jayswal (2005), Administering Data Centers 1st edition, Wiley

SEMESTER IV

ANDROID APPLICATION DEVELOPMENT WITH KOTLIN

Course Code	23U4VCBCA3
No. of credits	3
No. of contact hours	72

Course Description:

The aim of this course is to introduce to the students able to understand the process of developing application for the mobile and able to create the mobile applications on android platform in Kotlin which useful to the people in real life.

Course Outcomes (COs)

CO1	Design of applications using the concepts of Kotlin programming, android architecture, android components and user interface.
CO2	Design the android application using the View and ViewGroups
CO3	Implement the concept of shared preferences, Sqlite databases, content provider and maps in developing android applications
CO4	Design applications to store and retrieve data using the Firebase
CO5	Illustrate the concepts of networking, Retrofit library, Android JetPack and other libraries

Module 1: (15 Hours)

Introduction to KOTLIN Programming: Basics of Kotlin, Operations and Priorities, Decision Making Loop Control, Data Structures (Collections), Functions, Object Oriented Programming: Inheritance, abstract, interface, super and this, visibility modifiers.

Introduction to Android and User Interface: Android architecture and features, Tools, Android application components -Activity, Broadcast Receiver, Services, Intent, Introduction to manifest, externalizing resources, Android activities, activity lifecycle

Intents and their applications, Types of intents, Switching activities using intents, Sending data from one activity to another using intent, Working with implicit intents

Module 2: (12 Hours)

Introduction to XML layouts and different types of views, Button, TextView, ImageView, ProgressBar, ListView, EditText, Calendar, DateTime etc

Introduction and addition of action bar, Menus and dialogs, drawable and gradients, Introduction of recycle view and card view.

Module 3: (15 Hours)

SharedPreferences: saving and retrieving data from shared preferences

Content Provider: Creating content providers, Using content providers, Native Content providers

Android databases: Content values and cursors, Working with SQLite databases – CRUD operations

Maps and Location Based Services: Using location-based services, selecting a location provider, finding your current location, and creating map-based activities.

Module 4: (15 Hours)

Introduction to Firebase, authenticating users via Firebase, Using Firebase real-time database, storing data and retrieving data from Firebase

Introduction to SQLite Database-Using Room Persistence Library, storing data using Room database

Module 5: (15 Hours)

Introduction to Networking in Android, Using the Retrofit library, Working with REST APIs, Different types of network requests and how to implement them, JSON parsing, Discussion on different architecture patterns, Introduction to Android JetPack and advanced Android libraries, Signing and publishing applications, introduction to monetizing applications

Text Books:

1. Learn Android Studio 3 with Kotlin – Teg Hagos – Apress – 2019
2. Headfirst Kotlin, A Brain Friendly Guide – Dawn Griffiths, David Griffiths – Orilly – 2019
3. Learn Kotlin for Android Development – Peter Spath – Apress – 2019
4. Beginning Android Development With Kotlin - Greg Lim (Author) – March – 2020

References:

1. https://developer.android.com/kotlin

SEMESTER IV

MOBILE DEVICE AND NETWORK ARCHITECTURE

Course Code	23U4CRBCA13
No. of credits	3
No. of contact hours	72

Course Description:

Mobile traffic is growing and the need for more sophisticated broadband services will further push the limit on current standards to provide an even tighter integration between wireless technologies and higher speeds, requiring a new generation of mobile communications: This course aims to describe such features and technologies used in the evolution of mobile networks and architectures.

Course Outcomes (COs)

CO1	Describe the basic wireless communication principles and the wireless networks.
CO2	Explain the basic concepts of cellular networks and mobile handover with in the cellular network.
CO3	Illustrate the concepts of GSM, 2G, mobile IP and UMTS
CO4	Compare the important features and technologies used in the 3G, 4G and 5G networks.
CO5	Explain the features of a mobile device

Module 1: **(15 Hours)**

Communication Principles: Wireless Communication Principles, Radio Communication, Analog and Digital Communication, Benefits of Digital Signals, Computer Network, OSI Model, Mobile Network OSI layer functions, Mobile Network Protocol Layers. Introduction to Basic Telephony, POTS (Plain Old Telephone Service), Telephony Networks, PSTN (Public Switched Telephone Network), Telephone Network Hierarchy, Telecommunication Networks, Fixed Networks, Mobile Networks

Module 2: **(15 Hours)**

Mobile Cellular Networks Cellular Network Concepts, Cells and Base Stations, Frequency and Interface in Cells, Access Channels, Mobile Network Architecture, Mobile Network, Mobile Network Subsystems, Mobile Station, Base Station Subsystems, Network Switching Subsystems, Mobile Network Protocol Stacks, Core Networks, PLMN (Public Land Mobile Network), Mobile Network Fundamentals, Mobile Network Features, Mobility, Registration, Handoff, Roaming, Mobile Network Fundamentals (SMS), SMS (Short Message Service), SMS Network Architecture, SMS Network Elements, SMS Protocols, SMS Applications & Short Codes

Module 3: **(15 Hours)**

GSM and CDMA Networks: GSM History, GSM RF Channels, 2G Network Architecture, GSM Protocol Stack, GPRS Standards, CS and PS Domains, GPRS Architecture, GPRS Network Architecture, GPRS Protocols, CDMA Evolution, 2G CDMAOne, CDMA 2G Standards, 3GPP2 Network Architecture, Mobile IP

Module 4: **(15 Hours)**

UMTS Spectrum, UMTS Radio Access Network, UMTS Protocol Stacks, SIP Network, UMTS Multiple Access Network Architecture

4G TECHNOLOGY: IEEE 802.16 WiMax and LongTerm Evolution, OFDM, OFDMA, fourth generation cellular network, LTE Architecture: EPS/LTE Architecture, Evolved Packet Core (EPC), EPC Components: PDN Gateway (P-GW), Serving Gateway (S-GW), Mobility Management Entity (MME), Bearer Management: The EPS Bearer, Tunnelling Using GTP, Tunnelling Using GRE and PMIP, Signaling Radio Bearers.

Module 5: **(12 Hours)**

5G: ten pillars of 5G: Evolution of Existing RATs, Hyperdense Small-Cell Deployment, Self-Organising Network, Machine Type Communication, Developing Millimetre-Wave RATs, Redesigning Backhaul Links, Energy Efficiency, Allocation of New Spectrum for 5G, Spectrum Sharing, RAN Virtualisation, 5G Architecture.

Handset Characteristics and Features: Mobile Handset Categories, Low end Phones, Feature Phones, Smart phones, Handset Manufacture, Handset Bill of Materials, Handset hardware architecture, Primary Hardware Subsystems, Introduction and Definition to the SIM, Smartcards in general and concept of Mobile Identity, Functions and usage of the SIM, Phones without SIMs

References:

1. Yi-Bang Lin and Imrich Chlamtac, Wiley, Wireless and Mobile Network Architectures
2. Andre Perez, Wiley, Mobile Networks Architecture
3. Asoke. K Talukder, Roopa R. Yavagal, Asoke K. Talukder, Mobile Computing Technology Application & Service, Tata McGraw-Hill, 2020
4. Jörg Eberspächer, Hans-JoergV ögel, Christian Bettstetter, GSM Architecture, Protocols and Services, Christian Hartmann John Wiley & Sons
5. An introduction to LTE: LTE, LTE-advanced, SAE and 4G mobile communications, Christopher Cox, John Wiley & Sons
6. Fundamentals of 5G mobile networks, Edited by Jonathan Rodriguez, John Wiley & Sons, Ltd

SEMESTER IV

ANDROID APPLICATION DEVELOPMENT WITH KOTLIN– PRACTICAL

Course Code	23U4PVBCA1
No. of credits	2
No. of contact hours	54

Course Description:

The aim of this course is to introduce to the students able to understand the process of developing application for the mobile and able to create the mobile applications on android platform in Kotlin which useful to the people in real life.

Course Outcomes (COs)

CO1	Design of applications using the concepts of Kotlin programming, android architecture, android components and user interface.
CO2	Design the android application using the View and ViewGroups
CO3	Implement the concept of shared preferences, Sqlite databases, content provider and maps in developing android applications
CO4	Design applications to store and retrieve data using the Firebase
CO5	Illustrate the concepts of networking, Retrofit library, Android JetPack and other libraries

Continuous Internal Evaluation Pattern:

Attendance: 5 marks

Continuous Assessment: 20 Marks

End Semester Examination Pattern: 75 Marks

The following guidelines should be followed regarding award of marks

(a) Lab Programs: 35 Marks

(i) Algorithm/pseudocode: 5 Marks

(ii)Implementing the work/Conducting the experiment: 15 Marks (iii)Performance, result and inference (Usage of equipments and troubleshooting): 15 Marks

(b) Mini Project: 25 marks

(i) Implementation and demo: 15 Marks

(ii)Modification and results during the evaluation: 10 Marks

(c) Record: 5 Marks

(d) Viva Voce: 10 Marks

Mini Project is evaluated using the rubrics given below.

	Unacceptable	**Poor**	**Good**	**Excellent**
Solution	An incomplete solution is implemented on the required platform. It does not compile and /or run	A complete solution is implemented on the required platform and uses the compiler specified. It runs, but has logical errors	A completed solution is tested and runs but does not meet all the specifications and / or work for all test data	A completed solution runs without errors. It meets all the specifications and works for all test data.
Program Design	Few of the selected structures are appropriate. Program elements are not well designed	Not all of the selected structures are appropriate. Some of the program elements are appropriately designed	The program design generally uses appropriate structures. Program elements exhibit good design	The program design uses appropriate structures. The overall program design is appropriate
Interface	User interaction is incomplete and does not meet specifications	User interaction minimally meets the specifications, but does not increase the usability	User interaction generally meets the specifications, and is acceptable to the user	User interaction is as specified and is natural

List of Programs

1. Hello World
2. Square of an Integer
3. String Swapper
4. Simple Calculator
5. Unit Converter
6. Timetable

7. RadioButton, CheckBox Example
8. ImageView
9. Listview
10. RecyclerView
11. Intent -pass data
12. Message share by Implicit Intent
13. Email Send
14. SharedPreference - Profile
15. SharedPreference- Login
16. External Storage
17. SQLite Database, use DB Browser
18. Firebase

SEMESTER IV

INTRODUCTION TO CLOUD TECHNOLOGY – PRACTICAL

Course Code	23U4PVBCA2
No. of credits	1
No. of contact hours	36

Course Description:

To impart fundamentals concepts and applications in the area of cloud computing.

Course Outcomes (COs)

CO1	Understand the basics of cloud computing.
CO2	Know the Key concepts of Cloud Infrastructure Mechanisms.
CO3	Understand different Cloud Computing Architecture.
CO4	Understand the fundamentals of Cloud security.
CO5	Understand about cloud providers and case studies.

Continuous Internal Evaluation Pattern:

Attendance: 5 marks

Continuous Assessment: 20 Marks

End Semester Examination Pattern: 75 Marks

The following guidelines should be followed regarding award of marks

(a) Lab Programs: 75 Marks

(i) Algorithm/pseudocode: 10 Marks

(ii)Implementing the work/Conducting the experiment: 45 Marks

(iii)Performance, result and inference (Usage of equipment’s and troubleshooting): 20 Marks

List of Programs

1. Study the basic cloud architecture and represent it using a case study

2. Enlist Major difference between SAAS PAAS & IAAS also submit a research done on various companies in cloud business and the corresponding services provided by them, tag them under SAAS, PAAS & IAAS.
3. Present a report on obstacles and vulnerabilities in cloud computing on generic level
4. Present a report on Amazon cloud services.
5. Present a report on Microsoft cloud services.
6. Present a report on cost management on cloud
7. Enlist and explain legal issues involved in the cloud with the help of a case study
8. Explain the process of migrating to cloud with a case study.
9. Present a report on Google cloud and cloud services.

3.5 SEMESTR V

PRINCIPLES OF VIRTUALIZATION

Course Code	23U5VCBCA4
No. of credits	4
No. of contact hours	72

Course Description:

This course covers the basics of virtualization and virtualization software, deploying and managing an Enterprise Desktop Virtualization Environment, installation and implementation of Remote Desktop Services

Course Outcomes (COs)

CO1	Understand the basics of Virtualization
CO2	Discuss the various types of virtualization technologies
CO3	Explore the Application Virtualization technology and Remote desktop services
CO4	Understand the Virtualization products

Module 1: (15 Hours)

Origins and Influences, A Brief History, Definitions, Business Drivers, Technology Innovations: Clustering, Grid computing, Virtualization, Technology Innovations vs Enabling technologies, Understanding Virtualization, Important terms of Virtualization, Need of Virtualization, Benefits of Virtualization technology. Virtual Machine Basics, Taxonomy of Virtual Machines, Process Virtual Machines, System Virtual Machines, Hypervisor and its types, Major Hypervisor vendors.

Module 2: (20 Hours)

Virtualization Technologies: Server Virtualization, Storage Virtualization, I/O Virtualization, Network Virtualization, Client Virtualization, Application virtualization, Desktop virtualization, Understanding Virtualization Uses, Hardware Independence, Resource

Replication, Operating system based virtualization, Studying Server Consolidation: Development and Test Environments and Helping with Disaster Recovery, Hardware based Virtualization, Virtualization Management, Other Considerations.

Module 3: (20 Hours)

Overview of Application Virtualization, Advantages and Limitations of Application Virtualization, Prepare and Manage Remote Applications, Application Sharing, SharedAPPVNC, Installing and configuring Remote Application Server.

Azure Remoteapp, Remote Desktop Service, Remote Desktop service advantages and disadvantages, Remote Desktop Services components, Installing and configuring Remote Desktop Services on Windows, Remote Desktop Web Access, Working of RD web access, Configuring Role based application Provisioning.

Module 4: (17 Hours)

List of virtualization Software available. Vmware- VMWare Infrastructure Architecture, Introduction to Vsphere, ESXi, VCenter Server and Vsphere client. VMWare features. Microsoft Virtual Server: Introduction to HYPER-V role. Architecture of Hyper-, Features of Mircosoft Virtual Server, Hyper-v virtual networking, Use virtual Machine Snapshots. Monitor the performance of a Hyper-V server, Citrix XENDesktop fundamentals.

References:

1. William Von Hagen, Professional Xen Virtualization (2008), Wrox Publications, January, 2008.
2. James E. Smith, Ravi Nair, Virtual Machines (2005), Virtual Machines: Versatile Platforms for Systems and Processes, Elsevier/Morgan Kauffman.
3. Twan Grotenhuis, Rogier Dittner, Aaron Tiensivu, Ken Majors, Geoffrey Green, David Rule, Andy Jones, Matthijs ten Seldam (2006) Virtualization with Microsoft Virtual Server 2005, Syngress Publications.
4. Ivanka Menken, Gerard Blokdijk (2008) Virtualization the complete cornerstone guide to Virtualization best practices, Lightning Source Incorporated.
5. Chris Wolf, Erick M. Halter (2005) Virtualization:From the Desktop to the Enterprise.

SEMESTER V

SERVER OPERATING SYSTEM

Course Code	23U5VCBCA5
No. of credits	4
No. of contact hours	72

Course Description:

A server operating system is a type of operating system that is designed to be installed and used on a server computer. It is an advanced version of an operating system, having features and capabilities required within a client-server architecture or similar enterprise computing environment. The course aims to install Linux and Windows server operating systems and the implementation of different services.

Course Outcomes (COs)

CO1	Demonstrate the Installation and activation of Windows Server 2019 and configure the active directory domain services.
CO2	Configuration of DNS, DHCP services in Windows Server 2019.
CO3	Implementation of Hyper-V, Domain name service and Nano server in Windows Server 2019
CO4	Install RHEL 8 server and configure DHCP, squid, LDAP services
CO5	Configuration of ftp, web, DNS, SSH services in RHEL 8

Module 1: (15 Hours)

Install Windows Server 2019; install Windows Server 2019 features and roles; install and configure Windows Server Core, **Activation model:** Automatic Virtual Machine Activation (AVMA), Key Management Service (KMS), and Active Directory-based Activation,

Install and Configure Active Directory Domain Services: Install and configure domain controllers, forest, configure a read-only domain controller, Active Directory accounts: create, copy, configure, and delete user, Create and manage Active Directory groups.

Module 2: (15 Hours)

Networking with windows server 2019: install DNS, configure forwarders, configure Root Hints, configure delegation, implement DNS policies, create and configure DNS zones and records, DHCP, Create and manage Group Policy Objects.

Module 3: (12 Hours)

Hyper-V: virtualization modes, Hyper-V architecture, Hyper-V installation requirements, Hyper-V Manager, Configuring Hyper-V settings, **Nano Server**:_Install Nano Server, Implement Roles and Features on Nano Server, Manage and configure Nano Server, Containers**.** Security: bitlocker, server patching and updating.

Module 4: (15 Hours)

RHEL 8 installation, booting sequence, bash shell scripting, configuration of DHCP, samba share, web proxy: squid, LDAP, concepts of databases

Module 5: (15 Hours)

FTP server: NFS file systems and NFS exports, **web server**: configuring apache httpd, configuring and troubleshooting virtual hosts, concepts of SSL, TLS, DNS: hierarchy, DNS lookups, resource records and purposes, SSH, POP3, IMAP

References

1. Windows Server 2019 Administration Fundamentals Author: Bekim Dauti ISBN-10: 1788626567
2. Installation, Storage and Compute with Windows Server 2019 Author: Craig Zacker Published with the authorization of Microsoft Corporation by: Pearson Education, Inc.
3. Networking with Windows Server 2019 Author: Andrew Warren Published with the authorization of Microsoft Corporation by: Pearson Education, Inc.
4. Red Hat Enterprise Linux 8 Essentials: Learn to Install, Administer and Deploy RHEL 8 Systems, by Neil Smyth, 2019
5. Red hat Linux Administration By Michael Turner and Steve Shah-McGraw-Hill Companies,Inc Publisher, 201076
6. Redhat Fedora linux for Dummies By Naba Barkakati-Wiley Publishing, Inc, 2011
7. RHCSA/RHCE Red Hat Linux Certification Study Guide (Exams EX200 & EX300), 6th Edition (Certification Press) [Paperback], Michael Jang, McGraw-Hill Osborne Media; 6edition

SEMESTER V

IT AND ENVIRONMENT

Course Code	23U5CRBCA14
No. of credits	4
No. of contact hours	72

Course Description:

To understand legal and professional obligations regarding the environment and how information technology can contribute to reducing the environmental damage arising from human activities

Course Outcomes (COs)

CO1	Explain the main legislative provisions affecting the environmental impact of human activities.
CO2	Outline the need for a holistic approach in assessing environmental impact and be able to incorporate environmental considerations into a cost/benefit analysis
CO3	Explain the environmental impact of information systems and be able to draw up realistic plans for reducing this impact.
CO4	Discuss the range of applications of information technology that enable the environmental impact of human activity and natural changes to be monitored and possibly reduced;
CO5	Assess the potential for using information technology to reduce the environmental impact of specific activities

Module 1: (15 Hours)

Multidisciplinary nature of environmental studies: Definition, scope and importance, Need for public awareness. Natural Resources: Renewable and non-renewable resources: Natural resources and associated problems. a) Forest resources: Use and over-exploitation, deforestation, case studies. Timber extraction, mining, dams and their effects on forest and tribal people. b) Water resources: Use and over-utilization of surface and ground water, floods, drought, conflicts over water, dams-benefits and problems. c) Mineral resources: Use and exploitation, environmental effects of extracting and using mineral resources, case studies. d) Food resources: World food problems, changes caused by agriculture and overgrazing, effects of modern agriculture, fertilizer-pesticide problems, water logging, salinity, case studies. e) Energy resources: Growing energy needs, renewable and non-renewable energy sources, use of alternate energy sources, Case studies. f) Land resources: Land as a resource, land degradation, man induced landslides, soil erosion and desertification. Role of individual in conservation of natural resources. Equitable use of resources for sustainable life styles.

Ecosystems: Concept of an ecosystem, Structure and function of an ecosystem, Producers, consumers and decomposers, Energy flow in the ecosystem, Ecological succession, Food chains, food webs and ecological pyramids., Introduction, types, characteristic features, structure and function of the given ecosystem: Forest ecosystem

Module 2: (15 Hours)

Biodiversity and its conservation: Introduction, Bio-geographical classification of India, Value of biodiversity: consumptive use, productive use, social, ethical, aesthetic and option values. India as a mega-diversity nation, Hot-sports of biodiversity, Threats to biodiversity: habitat loss, poaching of wildlife, man-wildlife conflicts, Endangered and endemic species of India

Environmental Pollution: Definition, Causes, effects and control measures of: - Air pollution, Water pollution, Soil pollution, Marine pollution, Noise pollution, Thermal pollution, Nuclear hazards, Solid waste Management: Causes, effects and control measures of urban and industrial wastes., Role of an individual in prevention of pollution, Pollution case studies, Disaster management: floods, earthquake, cyclone and landslides.

Social Issues and the Environment: Urban problems related to energy, Water conservation, rain water harvesting, watershed management, Resettlement and rehabilitation of people: its problems and concerns, Case studies, Environmental ethics: Issues and possible solutions, Climate change, global warming, acid rain, ozone layer depletion, nuclear accidents and holocaust, Case studies, Consumerism and waste products, Environment Protection Act , Air (Prevention and Control of Pollution) Act, Water 91 (Prevention and control of Pollution) Act, Wildlife Protection Act, Forest Conservation Act, Issues involved in enforcement of environmental legislation, Public awareness.

Module 3: (15 Hours)

Internet as a knowledge repository, academic search techniques, creating cyber presence.

Academic websites, open access initiatives, opens access publishing models, Introduction to use of IT in teaching and learning- Educational software, Academic services–INFLIBNET, NPTEL, NICNET, BRNET.

Module 4: (15 Hours)

IT & Society- issues and concerns- digital divide, IT & development, the free software movement, IT industry: new opportunities and new threats, software piracy, cyber ethics, cyber-crime, cyber threats, cyber security, privacy issues, cyber laws, cyber addictions, information overload, health issues- guidelines for proper usage of computers, internet and mobile phones.

e-wastes and green computing, impact of IT on language & culture-localization issues- Unicode- IT and regional languages, Green Computing Concept

Module 5: (12 Hours)

Human Rights– An Introduction to Human Rights, Meaning, concept and development, Three Generations of Human Rights (Civil and Political Rights; Economic, Social and Cultural Rights). Human Rights and United Nations – contributions, main human rights related organs - UNESCO, UNICEF, WHO, ILO, Declarations for women and children, Universal Declaration of Human Rights. Human Rights in India – Fundamental rights and Indian Constitution, Rights for children and women, Scheduled Castes, Scheduled Tribes, Other Backward Castes and Minorities Environment and Human Rights - Right to Clean

Environment and Public Safety: Issues of Industrial Pollution, Prevention, Rehabilitation and Safety aspect of new technologies such as chemical and nuclear technologies, Issues of waste disposal, Protection of Environment Conservation of natural resources and human rights: reports, case studies and policy formulation. Conservation issues of western ghats- mention Gadgil committee report, kasthurirengan report. Over exploitation of ground water resources, marine fisheries, sand mining etc.

References:

1. Alan Evans, Kendall Martin, Mary Anne Poatsy Technology in Action, Pearson
2. Bharucha Erach (2021) Text Book of Environmental Studies for undergraduate Courses. 2nd Edition, University Press.
3. Clark.R.S Marine Pollution, Clanderson Press Oxford

4. Cunningham, W.P.Cooper, T.H.Gorhani, E & Hepworth, M.T (2001)Environmental Encyclopedia, Jaico Publication.
5. Dc A.K Environmental Chemistry, Wiley Eastern Ltd.
6. Heywood, V.H & Watson, R.T. (2021) Global Biodiversity Assessment, Cambridge University Press.

SEMESTER V

HYBRID MOBILE APPLICATION DEVELOPMENT

Course Code	23U5VCBCA6
No. of credits	3
No. of contact hours	72

Course Description:

The course familiarizes hybrid mobile application development using Flutter and React Native frameworks

Course Outcomes (COs)

CO1	Demonstrate the basics of flutter Hybrid framework and Dart Programming
CO2	Design flutter applications using widgets, theme and style, handle user inputs and gestures
CO3	Illustrate how to implement navigation and routing and to connect flutter application with the firebase
CO4	Develop hybrid mobile application using React Native framework
CO5	Illustrate integrating redux and Firebase to mobile application

Module 1: (15 Hours)

Flutter Application Development

Introduction: Different types of Mobile app development platforms: Native, Hybrid, Web applications, “Native” cross-platform apps, Advantages of hybrid apps.

Introduction to Dart Programming Language: Introduction to Dart, Writing Dart code DartPad, Installing Dart SDK, Installing Dart IDE and Writing Dart Program, understanding why Flutter uses Dart

Introduction to Flutter: Introduction, Comparisons with other mobile application frameworks, Key features of Flutter Hybrid framework, Setting up Flutter Environment, Flutter compilation. Flutter Rendering

Module 2: (15 Hours)

Flutter User Interface: Stateful verses Stateless Widgets, Built-in widgets- The Text Widget, The image widget, Material design and iOS Copertino widgets, Understanding Built-in Layout widgets, Creating UI with Widgets

Handling User inputs and Gestures: Handling user gestures, input widgets

Theming and Styling: Theme widgets, Material design

Module 3: (15 Hours)

Routing: Navigation between screens: understanding the navigator widget, named routes, screen transitions, hero animations

Developing fully featured apps: Setting up Firebase, Connecting Flutter app to firebase, Adding FlutterFire dependency to the Flutter project

Module 4: (15 Hours)

React Native: Key features of React Native Hybrid framework, Information flow, Architecture, Threading model. React Native benefits. Setting up an environment for developing iOS and Android apps, Introduction to JSX, Creating your First Application with create-react-app and Expo, Stateful versus presentational components, React lifecycle methods, The folder structure

React Native Basics: Styles, stylesheets, views and text, Components and state, Touch events, Text inputs, Buttons, The scrollview, Activity indicator, Working with images, Using modals

Navigation; Introduction, Installing navigation, Stack navigation Preview, working with Params, Navigation bar, Navigation buttons, The sidedrawer, More on the sidedrawer, Custom sidedrawer, Tab navigation, Nesting navigato

Working with libraries: Installing the image picker, Using the image picker

Module 5: (12 Hours)

Integrating redux and Firebase: Installing redux, Dispatching an action, Integrating firebase and creating forms, Register and loging with firebase

Debugging and Testing React Native: Debugging your React Native apps, Remote debugging, Logging, Inspecting React Native components. Testing: Introduction to the Jest testing framework, Snapshot testing your React Native components, working with functions, mocking modules

Text Books:

1. Alessandro Biessek, Flutter for Beginners, An Introductory guide to building cross-platform mobile applications with Flutter and Dart2, PACKT Publishing Limited
2. Akshat Paul, Abhishek Nalwaya, React Native for Mobile Development, Harness power of React Native to create stunning iOS and Android Applications, Apress Publishing Ltd

References:

1. Vladimir Novick, React Native - Building Mobile Apps with JavaScript, PACKT Publishing Limited, 2017, ISBN-13: 978-1787282537
2. Bonnie Eisenman, Learning React Native, 2nd Edition, Building Native Mobile Apps with JavaScript, O'Reilly Media Inc., 2017, ISBN-13: 978-9352136568
3. Eric Masiello, Jacob Friedmann, Mastering React Native, Packt Publishing, 2017, ISBN13: 978-1785885785
4. Stan Bershadskiy, Crysfel Villa, React Native Cookbook, Packt Publishing, 20 17, ISBN13: 978-1786462558
5. Emilio Rodriguez Martinez, React Native Blueprints: Create eight exciting native crossplatform mobile applications with JavaScript, Packt Publishing, 2017, ISBN-13: 978-1787288096
6. Flutter for Beginners: An introductory guide to building cross-platform mobile applications with Flutter
7. Web References: https://facebook.github.io/react-native/
   http://www.reactnative.com/
   https://www.flutter.dev/

SEMESTER V

SERVER OPERATING SYSTEM – PRACTICAL

Course Code	23U5PVBCA3
No. of credits	1
No. of contact hours	36

Course Description:

A server operating system (OS) is a type of operating system that is designed to be installed and used on a server computer. It is an advanced version of an operating system, having features and capabilities required within a client-server architecture or similar enterprise computing environment. The course aims to implement different services associated with Windows server as well as Linux environments.

Course Outcomes (COs)

COs	Course Outcomes Description
CO1	Installation of windows server 2019 operating systems virtually and setup environment.
CO2	Configure firewall, group policy objects, DNS, DHCP services for the administration of windows server 2019
CO3	Installation of RHEL 8 operating systems virtually and setup environment.
CO4	Configure DNS, FTP, web Server, DHCP in RHEL 8

Continuous Internal Evaluation Pattern:

Attendance: 5 marks

Continuous Assessment: 20 Marks

End Semester Examination Pattern: 75 Marks

The following guidelines should be followed regarding award of marks

(a) Lab Programs: 55 Marks

(i) Implementing the work/Conducting the experiment: 15 Marks

(ii)Performance, result and inference (Usage of equipment's and troubleshooting):

40 Marks

(b) Record: 10 Marks

(c) Viva Voce: 10 Marks

Part A: List of Programs in the Administration of Windows server 2019

1. Create a Virtual Install of windows server 2019 and windows server 2019 core.
2. Building a windows server 2019 Domain using PowerShell.
   a. Rename your server
   b. Assign static IP address
   c. Install Active Directory domain service using PowerShell
   d. Set server 2019 as Domain controller
   e. Add a windows client to the domain
3. Active Directory Administration
   a. Create a user in active directory
   b. Limit logon hours for a user in windows directory
   c. Limit computers a user can log onto
   d. Reset a user's password
   e. Unlock/enable an account in active directory
4. Server 2019 Administration
   a. Reset administrator password
   b. Create a shared folder using server manager
   c. Windows server update settings
   d. Enable/Disable remote Desktop
5. Group Policy
   a. Create and link a group policy to an OU
   b. Edit group policy objects to affect a user
   c. Edit group policy objects to affect a computer
   d. Backup and restore GPO in active directory
6. Domain Naming System
   a. Install DNS role using server manager
   b. Create a secondary or backup DNS server using a full install of Server 2019.

   c. Create a DNS reverse lookup zone in Server 2019.
   d. Create a new DNS host record in Server 2019.
   e. Create a new host record in Microsoft DNS for a new Website.
   f. Create a DNS MX record for a Microsoft Exchange server.
7. DHCP
   a. Install the DHCP Role Using PowerShell
   b. Installing the DHCP Server Role Using Server Manager
   c. How to authorize/Unauthorize a DHCP server
   d. How to Edit the default gateway settings in DHCP
   e. How to edit DNS server list in DHCP
   f. Edit a DHCP Scope
8. Firewall Administration
   a. Turn the Windows firewall on and off
   b. How to Allow an Inbound Port Thru Windows Firewall
   c. Using netsh to Manage the Windows Firewall
   d. Using PowerShell to Gather Firewall Information

Part B: List of Programs in the Administration of RHEL

1. Configure your system such that SE Linux must be in enforcing mode and firewall is enables and ssh service is not allowed through your firewall.
2. Configure ftp server such that anonymous can download and upload the data to ftp server. Deny users John and Carel to access the ftp server. Note that your ftp server must be accessible in your private network only. It can’t be accessible in another network.
3. Configure the DHCP server such that your DHCP server will able to provide IP configuration to 65 systems in your network.
4. Configure the station as NFS server such that /share directory is shared & only accessible in your network. This NFS share should be automatically mounted on remote client using autofs. On remote client system, NFS share should be mounted on /data/share directory.
5. Configure stationx.example.com for quota such that when user neo type
   dd if=/dev/zero of=/quota/somefile bs=1024 count=30, he succeed. When he type
   dd if=/dev/zero of=/quota/somefile bs=1024 count=70 he fails.

6. Implement a stationx.example.com as a web server for the site http://wwwx.rhce.com, then perform the following steps:
    a. Download ftp://server.rhce.com/pub/rhce/station.html.
    b. Rename the downloaded file to index.html.
    c. Copy this index.html to the DocumentRoot of your web server.
    d. Do not make any modification to the content of index.html.

SEMESTER V

HYBRID MOBILE APPLICATION DEVELOPMENT- PRACTICAL

Course Code	23U5PVBCA4
No. of credits	2
No. of contact hours	54

Course Description:

The course familiarizes hybrid mobile application development using Flutter and React Native frameworks

Course Outcomes (COs)

CO1	Understand the basics of Dart programming and flutter Hybrid framework
CO2	Implement flutter widget, navigation and routing
CO3	Illustrate how to connect flutter application with the firebase
CO4	Understand the basics of React Native Hybrid framework
CO5	Illustrate integrating redux and Firebase to React Native app

Continuous Internal Evaluation Pattern:

Attendance: 5 marks

Continuous Assessment: 20 Marks

End Semester Examination Pattern: 75 Marks

The following guidelines should be followed regarding award of marks

(a) Lab Programs: 35 Marks

(i) Algorithm/pseudocode: 5 Marks

(ii)Implementing the work/Conducting the experiment: 15 Marks

(iii)Performance, result and inference (Usage of equipment and troubleshooting): 15 Marks

(b) Mini Project: 25 marks

(i) Implementation and demo: 15 Marks

(ii)Modification and results during the evaluation: 10 Marks

(c) Record: 5 Marks

(d) Viva Voce: 10 Marks

Mini project is evaluated using the rubrics given below.

	Unacceptable	**Poor**	**Good**	**Excellent**
Solution	An incomplete solution is implemented on the required platform. It does not compile and /or run	A complete solution is implemented on the required platform and uses the compiler specified. It runs, but has logical errors	A completed solution is tested and runs but does not meet all the specifications and / or work for all test data	A completed solution runs without errors. It meets all the specifications and works for all test data.
Program Design	Few of the selected structures are appropriate. Program elements are not well designed	Not all of the selected structures are appropriate. Some of the program elements are appropriately designed	The program design generally uses appropriate structures. Program elements exhibit good design	The program design uses appropriate structures. The overall program design is appropriate
Interface	User interaction is incomplete and does not meet specifications	User interaction minimally meets the specifications, but does not increase the usability	User interaction generally meets the specifications, and is acceptable to the user	User interaction is as specified and is natural

Part A:

List of Programs: React Native Framework

1. Capturing Taps
2. Custom Component
3. State & Props
4. Styling
5. Scrollable Content
6. Building a form
7. Long Lists

8. Working with an API
9. Multiple files
10. Class components
11. Component Lifecycle

List of Programs: Flutter Framework

1. Flutter – Setting up Flutter environment
2. Flutter - Basic Application Example
3. Flutter Widgets
    i. Flutter Text
    ii. Flutter TextField
    iii. Flutter FlatButton
    iv. Flutter RaisedButton
    v. Flutter SnackBar
    vi. Flutter Switch
    vii. Flutter ToggleButtons
    viii. Flutter Table
    ix. Flutter DataTable
    x. Flutter Tooltip
    xi. Flutter TabBar & TabBarView
4. Flutter Animation
    i. Flutter Animation Basic Example
    ii. Flutter Animate Color
5. Flutter Packages
    i. Flutter sqlite - SQLite Tutorial
6. Flutter Examples
    i. Flutter Login Screen Sample

Part B: Mini Project

1. Mobile application using Flutter/ React Native framework

3.6 SEMESTER VI

DEVOPS FUNDAMENTALS

Course Code	23U6VCBCA7
No. of credits	3
No. of contact hours	54

Course Description:

DevOps is a process that enables continuous delivery of the services or products to the customers. It is a method that automates the development, process and operations to make sure that the software is developed, tested and delivered quickly with optimum reliability. This course is specially designed to introduce with DevOps; the professional movement the concentrates collaboration, communication, integration and automation to enhance the work flow among the software developers and IT operations professionals.

Course Outcomes (COs)

CO1	Describe the DevOps principles and practices in the area of software development and the project management.
CO2	Implement the version control during the software development process.
CO3	Familiarize the Docker containers including Application, Operating System, Database, programming languages.
CO4	Implement the container orchestration using kubernets
CO5	Explain the importance of automation process in DevOps with the help of ansible.

Module 1: (11 Hours)

DevOps and Project management: Agile vs DevOps, DevOps principles and Practices, DevOps and Software development process, DevOps and Project management.

Case study: with tools like Jira, Jenkins, Maven, AWS DevOps

Module 2: (11 Hours)

Version Control: Version control, benefits of version control, Use of Version Control System, key features, types of version control systems: local, centralized, distributed version control, selection of optimal version control system, use cases in each types, Update, Commit, push, pull. Common pitfalls in version control, Case study: Discus different version control tools such as Git, CVS, SVN

Module 3: (11 Hours)

Containerization: Containers, Container orchestration, benefits, types, virtualization vs containerization

Docker: features, component, installation in windows/Linux, Docker hub, Docker containers, Docker Architecture, multi container architecture in Docker, containers and hosts, containers and shells, Docker file, Docker networks, pushing images to remote repositories.

Case study: Prometheus - Monitoring system & time series database, Grafana: The open observability platform

Module 4: (11 Hours)

Container orchestration with Kubernets: Kubernets: introduction, architecture, different approaches of setting kubernet, namespaces, Objects, cluster setup in any of the platforms like AWS, terraform

Module 5: (10 Hours)

Automation: DevOps automation, Docker vs automation, automation tools and their key features

Ansible: Ansible architecture, ansible cluster formation, ansible adhoc commands and modules, ansible inventory grouping, playbooks, variables, loops, Debug module, Handlers, error handling tagging, ansible Vauit, ansible docker automation

References

1. The DevOps Adoption Playbook: A Guide to Adopting DevOps in a Multi-Speed IT Enterprise Published by John Wiley & Sons, Inc
2. DevOps for digital: leaders reignite business with a modern DevOps-enabled software factory by Aruna Ravichandran, Kieran Taylor, Peter Waterhouse
3. Ansible: From Beginner to Pro by Michael Heap, Apress publications
4. Red Hat Ansible Automation Platform: A beginner’s guide by Red Hat
5. Version Control with Git, Second Edition by Jon Loeliger and Matthew McCullough Published by O’Reilly Media, Inc.,
6. The Docker Book by James Turnbull

SEMESTER VI

INTRODUCTION TO MACHINE LEARNING AND DATA SCIENCE

Course Code	23U6CRBCA15
No. of credits	4
No. of contact hours	72

Course Description:

This course provides an introduction to the fundamental methods at the core of modern machine learning. It covers theoretical foundations as well as essential algorithms for supervised and unsupervised learning. This also provides details on data processing and representation.

Course Outcomes (COs)

CO1	Interpret data processing steps and data visualization techniques.
CO2	Demonstrate the supervised, unsupervised, semi-supervised, reinforcement Learning algorithms and the dimensionality reduction techniques
CO3	Implement the classification and the regression models using real world datasets
CO4	Construct the support vector classifier and the decision tree using real world datasets
CO5	Demonstrate the concepts of unsupervised learning and neural networks

Module 1: **(15 Hours)**

Knowing data, Data Pre-processing: Data cleaning, Data reduction, Data transformation, Data discretization Defining data visualization – Exploratory and Explanatory data visualization techniques Visualization workflow - Data Representation - chart types: categorical, hierarchical, relational, temporal & spatial; 2-D: bar charts, Clustered bar charts, dot plots, connected dot plots, pictograms, proportional shape charts, bubble charts, radar charts, polar charts, Range chart, Box-and-whisker plots, univariate scatter plots, histograms word cloud, pie chart, waffle chart, stacked bar chart, back-to-back bar chart, treemap. 3-D: surfaces, contours, hidden surfaces, pm3d coloring, 3D mapping; multi-dimensional data visualization; manifold visualization; graph data visualization; Annotation.

Module 2: **(15 Hours)**

What is Machine Learning? Machine Learning Vs. Traditional Programming, How Machine Learning Works? Applications of Machine Learning, Selecting the right features. Types of Learning – Supervised Learning, Unsupervised Learning, Semi-supervised Learning, Reinforcement Learning, Challenges in Machine Learning.

Probably Approximately Learning (PAC), Noise, Learning Multiple classes, Model Selection and Generalization, Dimensionality reduction- Subset selection, Principal Component Analysis

Module 3: **(15 Hours)**

Classifiers: Cross validation and resampling methods- Measuring classifier performance-Precision, recall. Bayes Theorem, Bayesian classifier, Maximum Likelihood estimation, Density functions, Random forest.

Regression: Types of Regression, Linear Regression, Multiple Linear Regression, Non-Linear Regression (Polynomial Regression). Optimizer: Basics, Definition , usage and types.

Module 4: **(15 Hours)**

Kernel Machine: Support Vector Machine- Optimal Separating hyperplane, Soft-margin hyperplane, Kernel trick

Decision Trees- Entropy, Information Gain, Tree construction, ID3, Issues in Decision Tree learning- Avoiding Overfitting, Reduced Error Pruning, The problem of Missing Attributes, Gain Ratio

Module 5: **(12 Hours)**

Unsupervised Learning - Clustering Methods - K-means, Hierarchical Clustering Methods, Density based clustering

Neural Networks- The Perceptron, Activation Functions, Training Feed Forward Network by Back Propagation. Introduction to Deep Learning.

Text Books:

1. Andy Kirk, Data Visualization A Handbook for Data Driven Design, Sage Publications, 2019

2. Cathy O’Neil and Rachel Schutt. Doing Data Science, Straight Talk From The Frontline. O’Reilly, 2019

References:

1. Jure Leskovek, Anand Rajaraman and Jeffrey Ullman. Mining of Massive Datasets. v2.1, Cambridge University Press.

SEMESTER VI

COMPUTER GRAPHICS USING OPENGL

Course Code	23U6CRBC16
No. of credits	3
No. of contact hours	54

Course Description:

The course covers fundamental topics such as graphics representations and transformations, and viewing pipeline.

Course Outcomes (COs)

CO1	Design and implement algorithms for 2D graphics primitives and attributes.
CO2	Illustrate Geometric transformations on both 2D and 3D objects.
CO3	Explain the concepts of clipping and visible surface detection in 2D and 3D viewing, and Illumination Models.
CO4	Implement computer graphics programs in Open GL

Module 1: (11 Hours)

Computer Graphics and OpenGL

Computer Graphics: Basics of computer graphics- pixel, resolution, aspect ratio, frame buffer, Application of Computer Graphics, Video Display Devices: Random Scan and Raster Scan displays

OpenGL: Introduction to OpenGL, coordinate reference frames, specifying two-dimensional world coordinate reference frames in OpenGL, OpenGL point functions, OpenGL line functions, point attributes, line attributes, curve attributes, OpenGL point attribute functions, OpenGL line attribute functions, Line drawing algorithms (DDA, Bresenham's), circle generation algorithms (Bresenham's).

Module 2: (11 Hours)

2D Geometric Transformations and 2D viewing: 2DGeometric Transformations: Basic 2D Geometric Transformations, matrix representations and homogeneous coordinates. Inverse transformations, 2DComposite transformations, other 2D transformations, raster methods for geometric transformations, OpenGL raster transformations, OpenGL geometric transformations function

2D viewing: 2D viewing pipeline, OpenGL 2D viewing functions.

Module 3: **(11 Hours)**

Clipping,3D Geometric Transformations

Clipping: clipping window, normalization and viewport transformations, clipping algorithms,2D point clipping 2D line clipping algorithms: cohen-sutherland line clipping only -polygon fill area clipping: Sutherland-Hodgeman polygon clipping algorithm only.

3DGeometric Transformations: 3D translation, rotation, scaling, composite 3D transformations, other 3D transformations, OpenGL geometric transformations functions.

Module 4: **(11 Hours)**

3D Viewing and Visible Surface Detection:

3DViewing:3D viewing concepts, 3D viewing pipeline, 3D viewing coordinate parameters, Transformation from world to viewing coordinates, Projection transformation, orthogonal projections, perspective projections, The viewport transformation and 3D screen coordinates. OpenGL 3D viewing functions.

Module 5: **(10 Hours)**

Visible Surface Detection Methods: Classification of visible surface Detection algorithms, back face detection, depth buffer method and OpenGL visibility detection functions.

Fill area Primitives: Polygon fill-areas, OpenGL polygon fill area functions, fill area attributes, general scan line polygon fill algorithm, OpenGL fill-area attribute functions.

Text Books:

1. Donald Hearn & Pauline Baker: Computer Graphics with OpenGL Version,3rd/ 4thEdition, Pearson Education,2011
2. Edward Angel: Interactive Computer Graphics- A Top Down approach with OpenGL, 5th edition. Pearson Education, 2008

References:

1. James D Foley, Andries Van Dam, Steven K Feiner, John F Huges Computer graphics with OpenGL: pearson education
2. Xiang, Plastock : Computer Graphics , sham's outline series, 2nd edition, TMG.
3. Kelvin Sung, Peter Shirley, steven Baer: Interactive Computer Graphics, concepts and applications, Cengage Learning
4. M MRaiker, Computer Graphics using OpenGL, Filip learning/Elsevier

SEMESTER VI

SOFTWARE LAB

Course Code	23U6PVBCA5
No. of credits	2
No. of contact hours	54

Course Description:

The course covers Computer Graphics using OpenGL and DevOps practical.

Part A: Computer Graphics using Open GL

Course Outcomes (COs)

CO1	Design and implement algorithms for 2D graphics primitives and attributes.
CO2	Illustrate Geometric transformations on both 2D objects.
CO3	Implement computer graphics programs in Open GL
CO4	Familiarization of Project management tools, Version Control tools, Containerization- Docker, Automation tools- Ansible

End Semester Examination Pattern: 75 Marks

The following guidelines should be followed regarding award of marks

(a) Lab Programs Part A: 30 Marks

(i) Algorithm/pseudocode: 5 Marks

(ii)Implementing the work/Conducting the experiment: 15 Marks

(iii)Performance, result and inference (Usage of equipment’s and troubleshooting): 10 Marks

(b) Lab Programs Part B: 30 Marks

(i) Algorithm/pseudocode: 5 Marks

(ii)Implementing the work/Conducting the experiment:15 Marks

(iii)Performance, result and inference (Usage of equipments and troubleshooting): 10 Marks

(c) Record: 5 Marks

(d) Viva Voice: 10 Marks

Continuous Internal Evaluation Pattern:

Attendance: 5 marks

Continuous Assessment: 20 Marks

List of Experiments

Part A: Computer Graphics using Open GL

1. DDA line algorithm
2. Bresenham's line drawing algorithm
3. Midpoint circle generation algorithm
4. Boundary fill algorithm
5. Translation algorithm
6. Rotation algorithm
7. scaling algorithm
8. reflection algorithm
9. shear algorithm
10. Ellipse Equation

Part B: DevOps Fundamentals

1. Familiarization of Project management tools
2. Familiarization of Version Control tools
3. Familiarization of Containerization- Docker
4. Familiarization of Automation tools- Ansible

SEMESTER VI

INTRODUCTION TO MACHINE LEARNING AND DATA SCIENCE - PRACTICAL

Course Code	23U6PRBCA7
No. of credits	1
No. of contact hours	36

Course Description:

This course contains the fundamentals on the standards and practices for collecting, organizing, managing, exploring, and using data. Topics include preparation, analysis, and visualization of data and creating analysis tools for larger data sets.

Course Outcomes (COs)

CO1	Interpret and implement data processing steps and data visualization techniques.
CO2	Interpret various learning approaches, and concepts of various machine learning techniques.
CO3	Implement classification methods and learning techniques.
CO4	Understand data distributions to obtain a decision.

End Semester Examination Pattern: 75 Marks

The following guidelines should be followed regarding award of marks

(a) Lab Programs: 55 Marks

(i) Algorithm/pseudocode: 5 Marks

(ii)Implementing the work/Conducting the experiment: 35 Marks

(iii)Performance, result and inference (Usage of equipments and troubleshooting): 15 Marks

(b) Record: 10 Marks

(c) Viva Voice: 10 Marks

Continuous Internal Evaluation Pattern:

Attendance: 5 marks

Continuous Assessment: 20 Marks

List of Experiments

2.1.1.1. K-means

2.1.1.2. KNN

2.1.1.3. Naïve bayes

2.1.1.4. Neural Network

2.1.1.5. Feed forward Neural Network with backpropagation

2.1.1.6. PCA

2.1.1.7. SVM

2.1.1.8. Decision tree with ID3 algorithm

2.1.1.9. Regression

SEMESTER VI

PROJECT AND VIVA-VOCE

Course Code	23U6PJBCA1
No. of credits	6
No. of contact hours	180

The BCA programme prepares the students to take up positions as programmers, System Analysts, System Designers in the field related to the computer science and information technology or students may go for higher studies in this area. The students are encouraged to involve themselves completely on the project work in their final semester. It is advised to students to develop their project for solving problems of software industry or any research organization. Doing this will give more exposure to handle real life problems of project development.

This project work is kept in BCA programme to give the opportunity to enhance skills in mobile application development on the Android platform and propel students towards the exciting opportunities in Android's future. Apart from developing android applications, students who are interested in adopting cloud resources or frameworks can do their project work on cloud computing aspects to gain hands-on practical experience to improve their competitiveness in the big data/cloud job market.

The project topic shall be chosen from areas of current day interest using latest packages/ languages running on appropriate platforms, so that the student can be trained to meet the requirements of the industry. This is an individual project or a group of at most three. The students can do project in any advanced language which is included in their syllabus.

A project report shall be submitted in hard bound and complete in all aspects. For internal evaluation, the progress of the student shall be systematically assessed through various stages of evaluation at periodic intervals.

4. OPEN COURSE

SECURITY THREATS AND TRENDS (OPEN COURSE)

Course Code	23U5OCBCA1
No. of credits	3
No. of contact hours	72

Course Description:

This course describes about security threats and trends, the basic concepts of Working of Internet, to understand the security threats and cyber laws, analyze the vulnerabilities of system and prevent it.

Course Outcomes (COs)

CO1	Explain the basic concepts of cyber security.
CO2	Summarize the security threats and analyze the vulnerabilities of system
CO3	Outline the basics of cryptography
CO4	Summarize the cyber-crimes and laws

Module 1: (18 Hours)

Internet and Basic working principle

Internet – Introduction, Basic Communication, Local Area Network, Packet Switching, Internet: A Network of Networks, ISPs and Network Connections, IP Address, Transmission Control Protocol (TCP), Domain Names. Internet Services: Electronic mail, browsing the World Wide Web, Automated Web Search (Search Engines), e-commerce.

Module 2: (18 Hours)

Understand about the viruses & worms

Introduction to Viruses &Worms**,** Working of Viruses & Worms, the various types of Viruses &Worms, managerial, technical & procedural controls to address Viruses & Worms. Introduction to Malware & Botnets, the concept of Honeynets and Honeypots, Remote Access Trojans & Rootkits, concepts, their working methods, Threats

&Information Warfare

Module 3: (18 Hours)

Prevention mechanism to secure a system

Authentication and Authorization; Public key encryption; Private key Encryption symmetric- key algorithms – DES, Triple DES, AES RSA; Tunneling and Encryption Protocols: IPSec; SSL and TLS; Firewall; Kerberos, SSL, SSH, Wireless Security.

Module 4: (18 Hours)

Cyber Crime and laws

Cyber Crimes – Computer Crime, Nature of Crimes, Penalty for damage to Computer, Computer system, Tampering with Computer Source Documents, Hacking, Computer Related Offences, Theft, The Language of Cyberspace.

References

1. Swiderski, Frank and Syndex, "Threat Modeling", Microsoft Press.
2. William Stallings and Lawrie Brown, "Computer Security: Principles and Practice", Prentice Hall.
3. Joseph M Kizza, "Computer Network Security", Springer Verlag
4. Thomas Calabres and Tom Calabrese, "Information Security Intelligence: Cryptographic Principles & Application", Thomson Delmar Learning.

5. MODEL QUESTION PAPERS

SEMESTER I

Reg. No .... Name **20U124**

B C A DEGREE END SEMESTER EXAMINATION - OCT 2020 : FEBRUARY 2023

SEMESTER 1 : MATHEMATICS - B C A (COMPLEMENTARY)

COURSE : **23U1CRCMT1/19U1CPCMT1 : FOUNDATION OF MATHEMATICS**

(For Regular - 2020 Admission and Supplementary/Improvement - 2019 Admission)

Time : Three Hours Max. Marks: 75

PART A

Answer any 10 (2 marks each)

1. What is the cardinality of the following sets:
a){{a}} and b){a}
2. Find f(S) if f(x)=[x^2/3] and S={-2,-1,0,1,2,3}.
3. Draw the graph of the function f(x)=[2x] from R to R.
4. Find $\bigcap_{i=1}^{n} \mathrm{A}_i \; and \; \bigcup_{i=1}^{n} \mathrm{A}_i$ if A_i={i,i+1,i+2, ... }.
5. List the ordered pairs in the relation R from A = {0, 1, 2, 3, 4} to B = {0,1,2,3}, where (a, b) ∈ R if and only if
a) a = b. b) a + b = 4.
c) a> b. d) a l b.
e) gcd (a, b) = 1. f) lcm (a, b) = 2.
6. Let R be be the relation R = {(a,b): a divides b} on the set of integers.
Find (a) R^{-1} (b) R
7. Suppose that the function f from A to B is a one-to-one correspondence. Let R be the relation that equals the graph of f. That is, R={(a,f(a)) : a ∈ A}. What is the inverse relation R^{-1}
8. a) Define a partial ordering.
b) Show that the divisibility relation on the set of positive integers is a partial order.
9. Draw the Truth Table for the Conditional Statement p → q.
10. Translate the sentence into logical expression: ”You will get an A in the class if and only if you either do every exercise in textbook or you get an A on the final ”.
11. Find the g.c.d of the pair of integers 58 and 86 and express it as a linear combination of the two integers.
12. Compute Φ (873)

(2 x 10 = 20)

PART B

Answer any 5 (5 marks each)

13. Define composition of functions .Let f and g be the functions from the set of integers to set of integers defined by f(x)=2x+3 and g(x)=3x+2.What is the composition of f and g?
14. What are the terms a_0,a_1,a_2,a_3 of the {an} where a_n=2^n+$(-2)^n$
15. For any sets,Prove that A-B=$A \cap \overline{B}$
16. These relations on the set of real numbers:
R1 = {(a, b) ∈ R2 | a > b}, the “greater than” relation,
R2 = {(a, b) ∈ R2 | a ≥ b}, the “greater than or equal to” relation,
R3 = {(a, b) ∈ R2 | a < b}, the “less than” relation,
R4 = {(a, b) ∈ R2 | a ≤ b}, the “less than or equal to” relation,
R5 = {(a, b) ∈ R2 | a = b}, the “equal to” relation,
R6 = {(a, b) ∈ R2 | a ≠ b}, the “unequal to” relation.

Reg. No .... **Name ....** **20U315**

B C A DEGREE END SEMESTER EXAMINATION - OCT. 2023: JANUARY 2023
SEMESTER 3: MOBILE APPLICATIONS AND CLOUD TECHNOLOGY
COURSE: 23U1CRBCA1: DIGITAL SYSTEM DESIGN
(Common for Regular - 2023 Admissions)

Time: Three Hours **Max. Marks: 75**

PART A

Answer any 8 (2 marks each)

1. Differentiate Minterms and Maxterms with an example
2. Prove the equation (x+y)(x+z) = x+yz
3. Explain the method of converting a hexadecimal number to decimal and give an example
4. List the basic laws of boolean algebra
5. Explain about Interrupt?
6. Simplify the expression AB+A(B+C)+B(B+C)
7. Explain the method for converting a decimal number to binary and octal with examples.
8. Discuss about Multiplexers and demultiplexers.
9. Describe about types of shift registers
10. Explain about the counters.

(2 x 8 = 16)

PART B

Answer any 7 (5 marks each)

11. Explain Logic gates and describe the working of various gates with truth tables

    Find a) 834+ 345 b) 632+167 c) 765-389 Using BCD
12. Draw the block diagram of ripple counter. Explain
13. Prove De Morgans theorem using truth table and Circuit
14. Perform the subtraction 110011-1111 using 1’s compliment and 2,s Compliment
15. Differentiate Synchronous and asynchronous counters
16. Explain Full adder circuit with diagram
17. Discuss about encoders and decoders
18. Explain the addressing modes.

19. Steps to convert SOP and POS to its standard normal form and convert the expressions given below

    F(A,B,C) = (A+B).(B+C).(A+C)

    F(A,B,C) = AC+AB+BC

(7 x 5 = 35)

PART C
Answer any 2 (12 marks each)

20. Explain odd Parity bit generator and checker using truth table and circuit diagram.
21. Explain XOR and its application
22. Explain about different types of flip flops and its working
23. Define K-MAP and don’t care condition

    Simplify using k-map

    F(N,X,Y,Z)=∑(0,1,2,3,4,6,7,11,15)

    F(X,Y,Z,W)=∑M(1,3,7,11,15) and dc(X,Y,Z,W)=∑M(0,2,5)

    F(a,b,c,d)=∑(2,3,6,7,8,10,11,13,14)

(12 x 2 = 24)

Reg. No .... Name 20U315

B C A DEGREE END SEMESTER EXAMINATION - OCT. 2023: JANUARY 2023
SEMESTER 4: MOBILE APPLICATIONS AND CLOUD TECHNOLOGY
COURSE: 23U1CRBCA2: INTRODUCTION TO PROGRAMMING
(Common for Regular - 2023 Admissions)

Time: Three Hours Max. Marks: 75

PART A

Answer any 8 (2 marks each)

1. Distinguish between variable and constant.
2. Differentiate between break and continue statement.
3. What is this pointer?
4. With the help of an example explain what is the use of typedef?
5. Explain array of objects.
6. What do you mean by new and delete operators? Explain with examples.
7. Explain the inline function
8. Explain class and object
9. Differentiate between private and protected modes?
10. List the excepts in C++

(2 x 8 = 16)

PART B

Answer any 7 (5 marks each)

11. Define structure and union. How does a union differ from a structure?
12. What is a function? What are the parts of function?
13. Differentiate between call by value and call by reference.
14. What is exception handling? Explain
15. What are the features of object oriented programming?
16. What is function overloading? Explain with examples.
17. What are command line arguments? Explain with an example
18. Explain about Inheritance.
19. Differentiate between different levels of Inheritance with an example program?

(7 x 5 = 35)

PART C

Answer any 2 (12 marks each)

20. Explain different loop control statements in C with example.
21. Bring out the salient features of structured programming and object programming?
22. Explain how files can be handled in c++. Explain various file opening modes and file pointers.
23. Explain the concepts of constructors and destructors?

(12 x 2 = 24)

SEMESTER II

Reg. No .... Name 21U243

B C A DEGREE END SEMESTER EXAMINATION - JULY 2023

SEMESTER 2: COMPLEMENTARY MATHEMATICS FOR B C A

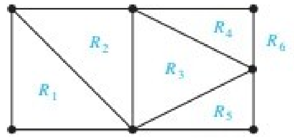
COURSE: **23U2CPCMT2; DISCRETE MATHEMATICS**

(Common for Regular 2020 Admission & Improvement/Supplementary 2019 Admission)

Time: Three Hours Max. Marks: 75

PART A
Answer any 10 (2 marks each)

1. What is the coefficient of x^7 in $(1 + x)^{11}$?
2. In how many ways can four mathematics books,three history books,three chemistry books and two sociology books be arranged on a shelf so that all books of the same subject are together?
3. Determine the coefficient of $w^2x^2y^2z^2$in the expansion of $(2w-x+3y+z-2)^{12}$?
4. (a) Define planar graph?
   (b) Is K_4 planar?
5. Find a Spanning tree for (a) $K_{1,6}$
   (b) K_4.
6. Verify Euler's formula for the following planar graph?



7. Explain Gauss-Seidel iteration method.
8. Use the method of iteration to solve the equation $x = e^{-x}$, starting with x_0 =1, correct to 3 decimal places.
9. Use Newton's method to find the root of $x^3 - 2x - 5 = 0$, correct to 2 decimal places with x_0 = 2.
10. Using Euler's method, find an approximate value of y corresponding to x = 1.2 , given that $\frac{dy}{dx} = x + 2y$ and y(1) = 1 taking grid size h = 0.1.
11. Solve $y' = -y$; y (0) = 1 by Euler's method for y(0.04).
12. Use Euler's method to approximate y when x = 0.1 given that $\frac{dy}{dx} = \frac{y-x}{y+x}$, y(0) = 1 by taking h = 0.05.

(2 x 10 = 20)

PART B
Answer any 5 (5 marks each)

13. In how many ways a football eleven can be chosen out of 17 players when (i) five particular players are to be always included?
    (ii) two particular players are to be always excluded?
14. How many words, with or without meaning,each of 3 vowels and 2 consonants can be formed from the letters of the word INVOLUTE?
15. If a connected planar simple graph has 20 vertices,each of degree 3.Into how many regions does a representation of this planar graph split the plane?
16. Describe Prim's algorithm ?
17. Solve the system of linear equations x + 2y + z = 8 ; 2x + 3y + 4z = 20 ; 4x + 3y + 2z = 16 using Gauss elimination method.

Reg. No .... **Name ....** **20U315**

B C A DEGREE END SEMESTER EXAMINATION - OCT. 2023: JANUARY 2023
SEMESTER 2: MOBILE APPLICATIONS AND CLOUD TECHNOLOGY
COURSE: 23U2CRBCA4: OPERATING SYSTEM
(Common for Regular - 2023 Admissions)

Time: Three Hours **Max. Marks: 75**

PART A

Answer any 5 (2 marks each)

1. Explain critical section problem.
2. Explain the terms first-fit, best fit, and worst fit.
3. Define is Semaphore? What are the different types of semaphore?
4. Define PCB with figure.
5. Explain different states of process.
6. Differentiate internal and external fragmentation?
7. List out different file operations
8. Explain about different directory structure.
9. Describe about LIFO, FIFO page replacement algorithm.
10. Explain the functions of operating system.

(2 x 8 = 16)

PART B

Answer any 7 (5 marks each)

11. Explain about IPC Mechanism.
12. Explain about Demand paging.
13. Define System calls. What are the different types of System Calls?
14. Explain Multiprogramming with diagram.
15. Explain Readers-writers problem.
16. Explain Resource allocation graphs.
17. Compare different page replacement algorithms.
18. Explain about segmentation.
19. Differentiate Symmetric key and Asymmetric key cryptography

(7 x 5 = 35)

PART C

Answer any 2 (12 marks each)

20. Explain deadlock. What are the necessary conditions of deadlock? Explain deadlock avoidance methods.
21. Explain Paging.
22. Explain CPU scheduling algorithms.
23. Explain the functions of operating system.

(12 x 2 = 24)

Reg. No .... Name 20U315

B C A DEGREE END SEMESTER EXAMINATION - OCT. 2023: JANUARY 2023
SEMESTER 2: MOBILE APPLICATIONS AND CLOUD TECHNOLOGY
COURSE: 23U2CRBCA5: PROBLEM SOLVING USING PYTHON
(Common for Regular - 2023 Admissions)

Time: Three Hours Max. Marks: 75

PART A

Answer any 5 (2 marks each)

1. What is the difference between tuples and lists in Python?
2. What is a Python dictionary?

3. Give the output of following Python code

```
str1='This  is Python'
print "Slice of String : ",str1[1:4]
print "Slice of String : ",str1[0:-1]
```

4. Write a lambda function to filter out all odd items from a list.
5. What is recursion?
6. What are packages in Python?
7. What is the purpose of readlines() method in Python?
8. Explain the modules in Python
9. Explain in detail about namespaces.
10. Describe about scoping in python

(2 x 8 = 16)

PART B

Answer any 7 (5 marks each)

11. Explain inheritance in Python
12. Explain how will you define and call a function in Python.
13. Explain variable length arguments in Python function.
14. Explain the use of Continue, break and pass keyword in python.
15. Explain different types of import statements in Python.
16. What is an exception? Explain with few examples.
17. Explain object orientation in python with example.
18. How do you count the number of times a value appears in a NumPy array of integers?
19. What is array slicing and how do you do it in NumPy?

(7 x 5 = 35)

PART C

Answer any 3 (10 marks each)

20. Explain for loop and for loop with else with explanation and example programs
21. Explain different modes for opening a file.
22. How to create Numpy arrays? List the advantages of Numpy over Python Lists
23. Write program to input two number and return their product. If the product is larger than 500, then return their sum?

(12 x 2 = 24)

Reg. No .... Name 20U315

B C A DEGREE END SEMESTER EXAMINATION - OCT. 2023: JANUARY 2023
SEMESTER 2: MOBILE APPLICATIONS AND CLOUD TECHNOLOGY
COURSE: 23U2CRBCA6: DATA STRUCTURES AND ALGORITHMS USING C++
(Common for Regular - 2023 Admissions)

Time: Three Hours Max. Marks: 75

PART A
Answer any 8 (2 marks each)

1. An array X[1..15][1..10] is stored in the memory with each element requiring 4 bytes of storage. If the base address of array is 1500, calculate the location of X[12][9] when the array X is stored in row major order?
2. What you mean by time complexity and space complexity?
3. Distinguish between static and dynamic memory allocation?
4. Give an algorithm to traverse a linked list?
5. Give a recursive algorithm for calculating Binomial Coefficient?
6. Define Circular linked list.
7. Explain how divide and conquer strategy is used in Quick Sort
8. Explain different operations on DEQUE
9. Explain how Circular Queue can be implemented?
10. Describe the basic operation on Circular Queue?

(2 x 8 = 16)

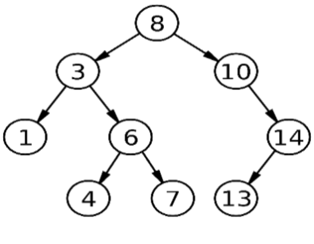
PART B
Answer any 7 (5 marks each)

11. Explain the classification of data structures with example. Explain recursive and non-recursive algorithms for binary search
12. Write a program to insert a node in Singly linked list in the below three cases
    i. At beginning
    ii. At end
    iii. After a specific node.
13. Explain basic operations of a stack and a queue.
14. Write a program to delete a node in Doubly linked list in the below three cases
15. Give an algorithm to delete node in a linked list?
16. Convert X: A+ (B*C-(D/E^F)*G)*H into postfix notation using stack.
17. Write a C program to sort N characters using insertion sort.
18. Explain Depth First Search with example.
19. Explain Breadth First Search with example

(7 x 5 = 35)

PART C
Answer any 3 (10 marks each)

20. Explain inorder, preorder and postorder traversal on a binary tree. Also trace how these traversals work on the following tree.



21. a) Give a recursive algorithm to traverse a Singly Linked List reversly.
    b) Explain three cases of deletion operation on Doubly Linked List
22. Write the algorithm for Merge Sort and trace Merge Sort algorithm on the list L= {78, 67, 90, 52, 82, 92, 33, 56, 18, 25}.
23. Explain different tree traversal algorithms

(12 x 2 = 24)

SEMESTER III

Reg. No .... Name **20U306**

B C A DEGREE END SEMESTER EXAMINATION - OCT. 2020: JANUARY 2021

SEMESTER 3 : MOBILE APPLICATIONS AND CLOUD TECHNOLOGY

COURSE : **23U3CRBCA7 : BASIC STATISTICS**

(Common for Regular - 2019 Admission & Supplementary 2018, 2017 & 2016 Admissions)

Time : Three Hours Max. Marks: 75

PART A

Answer All (1 mark each)

1. **Find the combined arithmetic mean of two samples of sizes 6 and 4 respectively whose means are 15 and 25.**
2. **Find the mean of the first six multiples of 4.**
3. **If the mean of 9, 8, 10, x, 12 is 15, find the value of x.**
4. **If the standard deviation of a data is 4.5 and if each value of the data is decreased by 5, then find the new standard deviation.**
5. **Prove that for two variables X and Y , Pearson's correlation coefficient rxy lies in the interval [-1,1].**
6. **If the standard deviation of a data is 3.6 and each value of the data is divided by 3, then find the new variance and new standard deviation.**
7. **Write down the Classical definition of probability.**
8. **State Multiplication theorem for three events.**
9. **What are the different types of Index numbers?**
10. **Explain Fisher's price Index.**

(1 x 10 = 10)

PART B

Answer any 8 (2 marks each)

11. **The mean yearly salary of employees of a company was Rs.36,000. The mean yearly salaries of male and female employees were Rs. 40,000 and Rs. 30,000 respectively.Find the percentage of male and female workers in the company.**
12. **The average height of 25 students was 160 cms. It was later found that the height of one student was misread as 120 cms. Instead of the correct value 180 cms. Calculate the correct average.**
13. **Calculate the median for the following data:**

Class:	**0-10**	**10-20**	**20-30**	**30-40**	**40-50**	**50-60**
Frequency:	**6**	**10**	**15**	**20**	**12**	**7**

14. **What are the properties of a good measure of dispersion?**
15. Write a short note on correlation and regression.
16. **Obtain the coefficient of quartile deviation for the following data:**

Class:	**0-10**	**10-20**	**20-30**	**30-40**	**4-50**	**50-60**	**60-70**
Freq:	**5**	**9**	**20**	**31**	**18**	**11**	**6**

17. **Find the probability of getting two heads when five coins are tossed.**
18. **An experiment results in one of the three mutually exclusive events A,B,and C. It is known that p(A) = 0.50 , P(B)= 0.30 and P(C)=0.20. Find the probabilities :**

    1. **$P(A \cap B)$ 2) $P(A \cup B)$ 3)$P(A/B)$**

19. **State and prove addition theorem for two events**
20. **Explain Time Series.**
21. **What are the objectives of time series?**
22. **What are the methods for finding trends?**

(2 x 8 = 16)

Reg. No .... **Name ....** **20U315**

B C A DEGREE END SEMESTER EXAMINATION - OCT. 2023: JANUARY 2023
SEMESTER 3: MOBILE APPLICATIONS AND CLOUD TECHNOLOGY
COURSE: 23U3CRBCA8: SOFTWARE ENGINEERING
(Common for Regular - 2023 Admissions)

Time: Three Hours **Max. Marks: 75**

PART A
Answer any 8 (2 marks each)

1. Discuss on waterfall model?
2. What are the advantages of evolutionary prototyping?
3. Distinguish between the Open-ended and Close-ended approach?
4. Why throw-away prototype should not be considered as a final system?
5. Explain about the architectural design process?
6. Explain about design heuristics for effective modularity?
7. What are the different classes of test cases of the regression test?
8. Explain the various design concepts considered during design?
9. Write short notes on the user interface design process?
10. Explain about risk analysis in software development.

(2 x 8 = 16)

PART B
Answer any 7 (5 marks each)

11. What are the fundamental activities of the software process?
12. The spiral model is one of the most effective life cycle models. Why?
13. Explore different feasibility studies to be conducted in the analysis?
14. Abstraction and refinement are complementary concepts in design. Explain?
15. Illustrate the different stages in a testing process?
16. Elaborate on the Software configuration management process?
17. What are the software maintenance cost factors?
18. Compute cyclomatic complexity.
19. Describe about COCOMO II Model.

(7 x 5 = 35)

PART C
Answer any 3 (10 marks each)

20. Explain different requirement elicitation techniques with its merits and demerits?

21. What are the various testing strategies to software testing? Discuss them briefly?

22. How CASE tools are classified. Explain about Software Cost estimation?

23. Draw the use case diagram for library management system.

(12 x 2 = 24)

Reg. No .... Name 20U315

B C A DEGREE END SEMESTER EXAMINATION - OCT. 2023: JANUARY 2023
SEMESTER 3: MOBILE APPLICATIONS AND CLOUD TECHNOLOGY
COURSE: 23U3CRBCA9: RDBMS
(Common for Regular - 2023 Admissions)

Time: Three Hours Max. Marks: 75

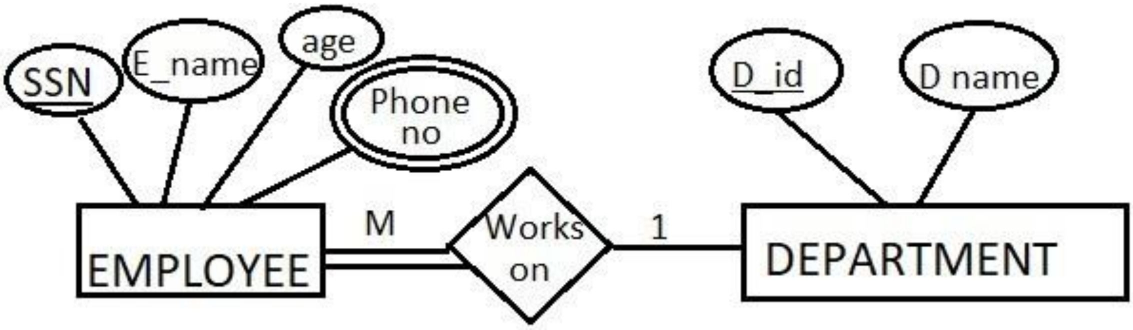
PART A
Answer any 8 (2 marks each)

1. Discuss the use of the transaction manager in DBMS?
2. Distinguish between the strong entity set and weak entity set?
3. Differentiate between primary and foreign key?
4. Describe the conditions to implement UNION operation in Relational Algebra.
5. Construct an SQL command to retrieve the list of employees and the projects they are working on, ordered by the department, and within each department, ordered alphabetically by the last name,then the first name.
6. Differentiate Functional dependency and Trivial functional dependency with examples.
7. What do you understand by database Normalization?
8. Explain different type of relationship exists between entities.
9. Illustrate ACID properties.
10. Define transitive functional dependency.

(2 x 8 = 16)

PART B
Answer any 7 (5 marks each)

11. Explain about the object based data models.
12. Construct an E-R diagram for a car-insurance company whose customers own one or more cars each.Each car has associated with it zero to any number of recorded accidents. Each insurance policy covers one or more cars, and has one or more premium payments associated with it. Each payment is for a particular period of time, and has associated due date, and the date when the payment wasreceived.
13. Discuss the characteristics of relations that make them different from ordinary tables and files?
14. Explain non-serial schedule.
15. What is transaction log? What are its functions?
16. Differentiate between the Share and the exclusive locks.
17. Convert the ER diagram given below to the relational model.



Find all relations from the above ER model. Write the relational schema for each relation

18. Consider the relation R=(ABCD) and the set of functional dependencies F=[AB→CD, C→A, D→B].Identify all the candidate keys of the relation R?

19. What is mean by lossless decomposition? illustrate with an example?

(7 x 5 = 35)

PART C

Answer any 2 (12 marks each)

20. UPS prides itself on having up-to-date information on the processing and current location of each shipped item. To do this, UPS relies on a company-wide information system. Shipped items are the heart of the UPS product tracking information system. Shipped items can be characterized by item number (unique), weight, dimensions, insurance amount, destination, and final delivery date. Shipped items are received into the UPS system at a single retail center. Retail centers are characterized by their type, uniqueID, and address. Shipped items make their way to their destination via one or more standard UPS transportation events (i.e., flights, truck deliveries). These transportation events are characterized by a unique scheduleNumber, a type (e.g, flight, truck), and a deliveryRoute. Create an Entity Relationship diagram that captures this information about the UPS system. Be certain to indicate identifiers and cardinality constraints.

21. Consider the following relations:
EMPLOYEE(SSN, NAME, GENDER, AGE, SALARY, DNUM)
DEPARTMENT(DNO, DNAME, DPHONE) PROJECTS(PNO, PNAME)
WORK_ON(SSN, PNO)
Construct the SQL expressions for retrieve the list of employees and the projects they are working on, ordered by department and within each department alphabetically by first name then last name.

22. What is normalization? Illustrate 1NF, 2NF, 3NF and BCNF with the help of example.

23. Explain recoverable and non-recoverable schedule with example?

(12 x 2 = 24)

Reg. No .... Name 20U315

B C A DEGREE END SEMESTER EXAMINATION - OCT. 2023: JANUARY 2023
SEMESTER 3: MOBILE APPLICATIONS AND CLOUD TECHNOLOGY
COURSE: 23U3CRBCA10: COMPUTER NETWORKS
(Common for Regular - 2023 Admissions)

Time: Three Hours Max. Marks: 75

PART A

Answer any 8 (2 marks each)

1.Describe the functions of Switch?

2.Describe Infrastructure mode wireless networks.

3.Describe message switching.

4.Describe briefly about IMAP?

5.What is ISDN?

6.What are the applications of leased lines?

7.Explain the networking features of LINUX operating system.

8.Differentiate between IPV4 addresses and IPV6 addresses

9.Explain the functions of network layer.

10. Explain the functions of ARP and RARP.

(2 x 8 = 16)

PART B

Answer any 7 (5 marks each)

11. Describe the different types of networks?
12. Differentiate between UTP and STP
13. Describe ICMP. Explain ICMP error messages.
14. Describe TCP/IP model.
15. Explain the types of wireless networks.
16. Describe about message switching with example
17. Explain about IGMP messages.
18. Describe IPV4 addressing.
19. Describe about Domain Controller.

(7 x 5 = 35)

PART C

Answer any 2 (12 marks each)

20. Explain in detail all the transmission media with its applications.
21. Describe the functions of the network devices in detail
22. Explain in detail about the protocols used for Email.
23. Describe the features and security in Windows NT Server. Explain in detail Windows NTdomain models.

(12 x 2 = 24)

Reg. No .... Name 20U315

B C A DEGREE END SEMESTER EXAMINATION - OCT. 2023: JANUARY 2023
SEMESTER 4: MOBILE APPLICATIONS AND CLOUD TECHNOLOGY
COURSE: 23U3CRBCA11: PROGRAMMING IN JAVA
(Common for Regular - 2023 Admissions)

Time: Three Hours Max. Marks: 75

PART A

Answer any 8 (2 marks each)

1. Define Java Virtual Machine.
2. Discuss about Java and World Wide Web.
3. When do we declare a method or class as final?
4. Differentiate between function overloading and function overriding.
5. What are the major difference between an interface and a class?
6. Define the benefits of a package.
7. Define about String class and its five methods with syntax.
8. Explain about java servlets
9. Illustrate the life cycle of a Thread.
10. Discuss about the Hibernet.

(2 x 8 = 16)

PART B

Answer any 7 (5 marks each)

11. Write a program for implementing multiple inheritance using interface.
12. What is an applet? Write the different states of an applet.
13. How does implement multiple inheritance in Java. Justify.
14. Explain in detail about the creation of a Thread in two ways.
15. Discuss about event class and event listeners.
16. Explain parameter passing mechanism.
17. Explain any five swing components.
18. Write a program to check a string is palindrome or not.
19. What is an array? What are the types of array supported in Java?

(7 x 5 = 35)

PART C

Answer any 3 (10 marks each)

20. Define exception handling. Explain more about the five terms which are used in exception handling.
21. Explain different types of layout managers.
22. Discuss about JDBC architecture, connection and statement.
23. Explain about event class and event listeners.

(12 x 2 = 24)

SEMESTER IV

Reg. No .... Name 20U315

B C A DEGREE END SEMESTER EXAMINATION - OCT. 2023: JANUARY 2023
SEMESTER 4: MOBILE APPLICATIONS AND CLOUD TECHNOLOGY
COURSE: 23U4CRBCA12: WEB TECHNOLOGIES
(Common for Regular - 2023 Admissions)

Time: Three Hours Max. Marks: 75

PART A
Answer any 8 (2 marks each)

1. Write an HTML program to create a table using the attributes cellspacing and cellpadding.
2. Write the code to show a video in HTML5.
3. Define the advantages of CSS and it syntax.
4. Differentiate between <*P*> and <*Pre*> tags.
5. Discuss about Unordered List. What are the different values of *list-style-type*?
6. Write a Java Script to find the sum of two numbers using user-defined function.
7. Differentiate between *Get* and *Post* methods.
8. Describe about jQuery.
9. Explain about the javascript arrays.
10. Differentiate HTML and XML

(2 x 5 = 16)

PART B
Answer any 7 (5 marks each)

11. What are the uses of formatting tags? Write any 5 formatting tags in detail.
12. Define DHTML and XHTML.
13. Discuss in detail about the three ways to insert CSS.
14. Discuss about rendering templates in Flask.
15. Define DOM.
16. Write a PHP program code to check whether the given number is prime or not.
17. What is session variable? Write the built-in function for starting a session.
18. Discuss about arrays in PHP
19. Define about containers in Python.

(7 x 5 = 35)

PART C
Answer any 2 (12 marks each)

20. Discuss about Ajax - Introduction, advantages & disadvantages, Purpose.
21. Explain three types of PHP – MySQL connection with example.
22. How does Flask useful for web development?
23. Write a Java Script program to validate username and password

(12 x 2 = 24)

Reg. No .... Name 20U315

B C A DEGREE END SEMESTER EXAMINATION - OCT. 2023: JANUARY 2023
SEMESTER 4: MOBILE APPLICATIONS AND CLOUD TECHNOLOGY
COURSE: 23U4VCBCA1: INTRODUCTION TO CLOUD TECHNOLOGY
(Common for Regular - 2023 Admissions)

Time: Three Hours Max. Marks: 75

PART A

Answer any 8 (2 marks each)

1. What are the essential characteristics of Cloud computing?
2. Define Cloudnomics.
3. What are the different types of cloud?
4. What are efficient steps that is needed for migrating into the cloud?
5. Explain the different Cloud Mitigation Strategies?
6. List the major Cloud components in Cloud Infrastruture
7. What are the obstacles for Cloud Technology?
8. List the different considerations for selecting the Cloud Solutions.
9. Which are the different layers that define cloud architecture?
10. Explain about IaaS.

(2 x 8 = 16)

PART B

Answer any 7 (5 marks each)

11. Describe the best practices used in selection of Cloud Provider
12. Explain the different actors involved in the Cloud computing Reference Model
13. Write short note on Cloud Security Challenges?
14. Explain different risk assessment methods in cloud computing
15. Explain in detail about the seven-step model used in cloud computing
16. Explain in detail about the deployment models in cloud computing
17. Explain and illustrate with different case studies of migrating into cloud
18. Describe the three common approaches to migrate the on-premise applications to cloud
19. Explain the need for Cloud Governance

(7 x 5 = 35)

PART C

Answer any 2 (12 marks each)

20. Explain the Cloud Computing Reference Architecture Model in detail with a neat diagram.
21. Explain the Seven- step Model of Migration into a Cloud.
22. Explain the ten cloud do and do nots in Cloud Technology.
23. Differentiate between provider side and client-side encryption standards

(12 x 2 = 24)

SEMESTER V

Reg. No .... Name 20U315

B C A DEGREE END SEMESTER EXAMINATION - OCT. 2023: JANUARY 2023
SEMESTER 5: MOBILE APPLICATIONS AND CLOUD TECHNOLOGY
COURSE: 23U5VCBCA4: PRINCIPLES OF VIRTUALIZATION
(Common for Regular - 2023 Admissions)

Time: Three Hours Max. Marks: 75

PART A
Answer any 8 (2 marks each)

1. How can the virtualisation play a major role in Software Development and testing?
2. Discuss VMware Consolidated Backup?
3. List out the different ways to allocate the system resources to a Virtual Machine?
4. What is the drawback of creating virtual machines with available default configuration?
5. What are the different types of application Virtualization? Explain.
6. Outline the functionality of Remote application server?
7. Why do we need Client Access Licence (CAL) in RD Licensing?
8. What is RDS?
9. List the steps to install and configure Remote Desktop Services on Windows.
10. Explain about PCI Controller.

(2 x 8 = 16)

PART B
Answer any 7 (5 marks each)

11. Explain the challenges encountered in Disaster Recovery?
12. Compare the differences between ESX and GSX servers.
13. How do you configure Virtual Machine Resources?
14. Compare and discuss the SCSI controller, and PS2 controller in Virtual Network?
15. Illustrate the steps to install Remote Desktop Services on Windows?
16. Discuss on Vmware Vcenter Server?
17. Discuss the different types of Hyper-V Virtual Networks?
18. Distinguish between vSphere Client and vCenter Server?
19. How can we monitor the overall performance of the network?

(7 x 5 = 35)

PART C
Answer any 2 (12 marks each)

20. Explain storage, desktop and network virtualization along with its advantages.
21. Explain how to configure BIOS to support Hardware Virtualization?
22. Discuss the major components of Citrix XenDesktop?
23. Explain about the commonly used tools for application and file sharing between computers.

(12 x 2= 24)

SEMESTER VI

Reg. No .... Name 20U315

B C A DEGREE END SEMESTER EXAMINATION - OCT. 2023: JANUARY 2023
SEMESTER 6: MOBILE APPLICATIONS AND CLOUD TECHNOLOGY
COURSE: 23U6CRBCA15: COMPUTER GRAPHICS USING OPENGL
(Common for Regular - 2023 Admissions)

Time: Three Hours Max. Marks: 75

PART A

Answer any 8 (2 marks each)

1. Define Computer graphics. State its significance in image processing.
2. Define the operation translation of an image.
3. Discuss the significance of frame buffer.
4. Define the term pixel. State how its related to resolution of an image.
5. Write the matrix representation of 2D rotation.
6. Why homogeneous coordinates are used in computer graphics.
7. State the condition where clipping is required
8. Explain about video display devices.
9. Differentiate horizontal retrace and vertical retrace.
10. Write the transformation function for rotation, scaling.

(2 x 8 = 16)

PART B

Answer any 7 (5 marks each)

11. Explain the drawbacks of Bresenham’s line drawing algorithm.
12. Make a short note on applications of computer graphics.
13. State the difference between open GL point function and line function.
14. Define scaling. Write a program to scale an object.
15. Explain about 2D composite transformation.
16. Make a short note on 2D viewing pipeline.
17. With neat and labelled diagram explain how a 3D image is translated.
18. Explain any 4 open GL homogeneous geometric transformation functions.
19. Compare and contrast orthogonal and perspective projection.

(7 x 5 = 35)

PART C

Answer any 2 (12 marks each)

20. Prepare a detailed note on visible surface detection methods.
21. Explain the working of DDA line drawing algorithm with example.
22. Discuss any two 2D basic transformations.
23. Explain Cohen Sutherland line clipping algorithm with example.

(12 x 2 = 24)