B.Sc. Computer Science (Cloud Computing and Cyber Security)
(Semester Pattern)

CHOICE BASED CREDIT SYSTEM REVISED SYLLABUS
(With effect from 2023-24)

1. **Course objectives :**
   - To prepare the students to manage the software components in a computer independently and to be a Programmer.
   - To motivate the students to take up higher studies in Computer Science and other streams.
2. **Eligibility for Admission:**

   A candidate should have studied +2 level Mathematics as one ofthesubjects in the 10 +2 stream.
3. **Duration of the Course:**

   The students shall undergo the prescribed course of study for a period of not less than three academic years (Six semesters).
4. **Medium of Instruction:** English.
5. **Eligibility for the Degree:**
   - A Candidate shall be eligible for the award of the degree on completion of the prescribed course of study and passing all the prescribed external examinations.
   - Attendance progress, internal examinations, conduct certificate from the Head of the Institution shall be required for taking the external examination.
   - The passing minimum and the ranking are as per the existing rule of the Choice Based Credit System for the affiliated college of the University.

6. Consolidated Semester wise and Component wise Credit distribution

Parts	Sem I	Sem II	Sem III	Sem IV	Sem V	Sem VI	Total Credits
Part I	3	3	3	3	-	-	12
Part II	3	3	3	3	-	-	12
Part III	13	13	13	13	22	18	92
Part IV	4	4	3	6	4	2	23
Part V	-	-	-	-	-	1	1
Total	23	23	22	25	26	21	**140**

***Part I. II , and Part III components will be separately taken into account for CGPA calculation and classification for the under graduate programme and the other components. IV, V has to be completed during the duration of the programme as per the norms, to be eligible for obtaining the UG degree.**

MethodsofEvaluation		
Internal Evaluation	ContinuousInternalAssessmentTest	25 Marks
	Assignments	
	Seminars	
	AttendanceandClassParticipation	
External Evaluation	EndSemesterExamination	75 Marks
	Total	100 Marks
MethodsofAssessment		
Recall(K1)	Simpledefinitions,MCQ,Recallsteps,Conceptdefinitions	
Understand/Comprehend(K2)	MCQ,True/False,Shortessays,Conceptexplanations,Shortsummaryor Overview	
Application (K3)	Suggestidea/conceptwithexamples,Suggestformulae, Solveproblems, Observe,Explain	
Analyze(K4)	Problem-solvingquestions,Finishaprocedureinmanysteps,Differentiate	
	betweenvariousideas,Mapknowledge	
Evaluate(K5)	Longer essay/Evaluationessay,Critiqueorjustifywithprosandcons	
Create(K6)	Checkknowledgeinspecificoroffbeatsituations,Discussion,Debatingor Presentations	

1. Introduction

B.Sc. Computer Science(cloud computing and cyber security)

Education is the key to development of any society. Role of higher education is crucial for securing right kind of employment and also to pursue further studies in best available world class institutes elsewhere within and outside India. Quality education in general and higher education in particular deserves high priority to enable the young and future generation of students to acquire skill, training and knowledge in order to enhance their thinking, creativity, comprehension and application abilities and prepare them to compete, succeed and excel globally. Learning Outcomes-based Curriculum Framework (LOCF) which makes it student-centric, interactive and outcome-oriented with well-defined aims, objectives and goals to achieve. LOCF also aims at ensuring uniform education standard and content delivery across the state which will help the students to ensure similar quality of education irrespective of the institute and location.

Computer Science is the study of quantity, structure, space and change, focusing on problem solving, application development with wider scope of application in science, engineering, technology, social sciences etc. throughout the world in last couple of decades and it has carved out a space for itself like any other disciplines of basic science and

engineering. Computer science is a discipline that spans theory and practice and it requires thinking both in abstract terms and in concrete terms. Nowadays, practically everyone is a computer user, and many people are even computer programmers. Computer Science can be seen on a higher level, as a science of problem solving and problem solving requires precision, creativity, and careful reasoning. The ever-evolving discipline of computer science also has strong connections to other disciplines. Many problems in science, engineering, health care, business, and other areas can be solved effectively with computers, but finding a solution requires both computer science expertise and knowledge of the particular application domain. Computer science has a wide range of specialties. These include Computer Architecture, Software Systems, Graphics, Artificial Intelligence, Computational Science, and Software Engineering. Drawing from a common core of computer science knowledge, each specialty area focuses on specific challenges. Computer Science is practiced by mathematicians, scientists and engineers. Mathematics, the origins of Computer Science, provides reason and logic. Science provides the methodology for learning and refinement. Engineering provides the techniques for building hardware and software.

Programme Outcome, Programme Specific Outcome and Course Outcome

Computer Science is the study of quantity, structure, space and change, focusing on problem solving, application development with wider scope of application in science, engineering, technology, social sciences etc. The key core areas of study in Mathematics include Algebra, Analysis (Real & Complex), Differential Equations, Geometry, and Mechanics. The

Students completing this programme will be able to present Software application clearly and precisely, make abstract ideas precise by formulating them in the Computer languages. Completion of this programme will also enable the learners to join teaching profession, enhance their employability for government jobs, jobs in software industry, banking, insurance and investment sectors, data analyst jobs and jobs in various other public and private enterprises.

2. Programme Outcomes (PO) of B.Sc. degree programme in Computer Science(cloud computing and cyber security)

- Scientific aptitude will be developed in Students
- Students will acquire basic Practical skills & Technical knowledge along with domain knowledge of different subjects in the Computer Science & humanities stream.
- Students will become employable; Students will be eligible for career opportunities in education field, Industry, or will be able to opt for entrepreneurship.

- Students will possess basic subject knowledge required for higher studies, professional and applied courses.
- Students will be aware of and able to develop solution oriented approach towards various Social and Environmental issues.
- Ability to acquire in-depth knowledge of several branches of Computer Science and aligned areas. This Programme helps learners in building a solid foundation for higher studies in Computer Science and applications.
- The skills and knowledge gained leads to proficiency in analytical reasoning, which can be utilized in modelling and solving real life problems.
- Utilize computer programming skills to solve theoretical and applied problems by critical understanding, analysis and synthesis.
- To recognize patterns and to identify essential and relevant aspects of problems.
- Ability to share ideas and insights while seeking and benefitting from knowledge and insight of others.
- Mould the students into responsible citizens in a rapidly changing interdependent society.

The above expectations generally can be pooled into 6 broad categories and can be modified according to institutional requirements:

PO1: Knowledge

PO2: Problem Analysis

PO3: Design / Development of Solutions

PO4: Conduct investigations of complex problems

PO5: Modern tool usage

PO6: Applying to society

3. Programme Specific Outcomes of B.Sc. Degree Programme in Computer Science(cloud computing and cyber security)

PSO1: Think in a critical and logical based manner

PSO2: Familiarize the students with suitable software tools of computer science and industrial applications to handle issues and solve problems in mathematics or statistics and realtime application related sciences.

PSO3: Know when there is a need for information, to be able to identify, locate, evaluate, and effectively use that information for the issue or problem at hand.

PSO4: Understand, formulate, develop programming model with logical approaches to a Address issues arising in social science, business and other contexts.

PSO5: Acquire good knowledge and understanding to solve specific theoretical and applied problems in advanced areas of Computer science and Industrial statistics.

PO6: Provide students/learners sufficient knowledge and skills enabling them to undertake further studies in Computer Science or Applications or Information Technology and its allied areas on multiple disciplines linked with Computer Science.

PO7: Equip with Computer science technical ability, problem solving skills, creative talent and power of communication necessary for various forms of employment.

PO8: Develop a range of generic skills helpful in employment, internships& societal activities.

PO9: Get adequate exposure to global and local concerns that provides platform for further exploration into multi-dimensional aspects of Computing sciences.

Mapping of Course Learning Outcomes (CLOs) with Programme Outcomes (POs) and Programme Specific Outcomes (PSOs) can be carried out accordingly, assigning the appropriate level in the grids: (put tick mark in each row)

4. Highlights of the Revamped Curriculum

- Student-centric, meeting the demands of industry & society, incorporating industrial components, hands-on training, skill enhancement modules, industrial project, project with viva-voce, exposure to entrepreneurial skills, training for competitive examinations, sustaining the quality of the core components and incorporating application oriented content wherever required.
- The Core subjects include latest developments in the education and scientific front, advanced programming packages allied with the discipline topics, practical training, devising mathematical models and algorithms for providing solutions to industry / real life situations. The curriculum also facilitates peer learning with advanced mathematical topics in the final semester, catering to the needs of stakeholders with research aptitude.
- The General Studies and Computer Science based problem solving skills are included as mandatory components in the „Training for Competitive Examinations" course at the final semester, a first of its kind.
- The curriculum is designed so as to strengthen the Industry-Academia interface and provide more job opportunities for the students.
- The Industrial Statistics course is newly introduced in the fourth semester, to expose

the students to real life problems and train the students on designing a mathematical model to provide solutions to the industrial problems.

- The Internship during the second year vacation will help the students gain valuable work experience that connects classroom knowledge to real world experience and to narrow down and focus on the career path.
- Project with viva-voce component in the fifth semester enables the student, application of conceptual knowledge to practical situations. The state of art technologies in conducting a Explain in a scientific and systematic way and arriving at a precise solution is ensured. Such innovative provisions of the industrial training, project and internships will give students an edge over the counterparts in the job market.
- State-of Art techniques from the streams of multi-disciplinary, cross disciplinary and inter disciplinary nature are incorporated as Elective courses, covering conventional topics to the latest – Statistics with R Programming, Data Science, Machine learing. Internet of Things and Artificial Intelligence etc..

5. Value additions in the Revamped Curriculum:

Semester	**Newly introduced Components**	**Outcome / Benefits**
I	**Foundation Course** To ease the transition of learning from higher secondary to higher	• Instil confidence among students • Create interest for the subject
	education, providing an overview of the pedagogy of learning abstract Mathematics and simulating mathematical concepts to real world.	
I, II, III, IV	**Skill Enhancement papers** (Discipline centric / Generic / Entrepreneurial)	• Industry ready graduates • Skilled human resource • Students are equipped with essential skills to make them employable
		• Data analytical skills will enable students gain internships, apprenticeships, field work involving data collection, compilation, analysis etc.

		• Entrepreneurial skill training will provide an opportunity for independent livelihood • Generates self – employment • Create small scale entrepreneurs • Training to girls leads to women empowerment
		• Discipline centric skill will improve the Technical knowhow of solving real life problems using ICT tools
III, IV, V & VI	Elective papers- An open choice of topics categorized under Generic and Discipline Centric	• Strengthening the domain knowledge • Introducing the stakeholders to the State-of Art techniques from the streams of multi-disciplinary, cross disciplinary and inter disciplinary nature • Students are exposed to Latest topics on Computer Science / IT, that require strong mathematical background • Emerging topics in higher education / industry / communication network / health sector etc. are introduced with hands-on-training, facilitates designing of mathematical models in the respective sectors
IV	Industrial Statistics	• Exposure to industry moulds students into solution providers • Generates Industry ready graduates • Employment opportunities enhanced
II year Vacation activity	Internship / Industrial Training	• Practical training at the Industry/ Banking Sector / Private/ Public sector organizations / Educational institutions, enable the students gain professional experience and also become responsible citizens.
V Semester	Project with Viva – voce	• Self-learning is enhanced • Application of the concept to real situation is conceived resulting in tangible outcome
VI Semester	Introduction of Professional Competency component	• Curriculum design accommodates all category of learners; „Mathematics for Advanced Explain" component will comprise of advanced topics in Mathematics and allied fields, for those in the peer group / aspiring researchers; • „Training for Competitive Examinations" –caters to the needs of the aspirants towards most sought - after services of the nation viz, UPSC, CDS, NDA, Banking Services, CAT, TNPSC group services, etc.

Extra Credits: **For Advanced Learners / Honors degree**	• To cater to the needs of peer learners / research aspirants

Skills acquired from the Courses	Knowledge, Problem Solving, Analytical ability, Professional Competency, Professional Communication and Transferrable Skill

Credit Distribution for UG Programmes

Sem I	Credit	H	Sem II	Credit	H	Sem III	Credit	H	Sem IV	Credit	H	Sem V	Credit	H	Sem VI	Credit	H
Part 1. Language – Tamil	3	6	Part..1. Language – Tamil	3	6	Part..1. Language – Tamil	3	6	Part..1. Language – Tamil	3	6	5.1 Core Course – \CC IX	4	5	6.1 Core Course – CC XIII	4	6
Part.2 English	3	6	Part.. 2 English	3	6	Part..2 English	3	6	Part.. 2 English	3	6	5.2 Core Course – CC X	4	5	6.2 Core Course – CC XIV	4	6
1.3 Core Course – CC I	5	5	2..3 Core Course – CCIII	5	5	3.3 Core Course – CC V	5	5	4.3 Core Course – CCVII Core Industry Module	5	5	5. 3.Core Course CC - XI	4	5	6.3 Core Course – CC XV	4	6
1.4 Core Course – CCII	5	5	2.4 Core Course – CCIV	5	5	3.4 Core Course – CC VI	5	5	4.4 Core Course – CC VIII	5	5	5. 4.Core Course –/ Project with viva-voce CC -XII	4	5	6.4 Elective -VII Generic/ Discipline Specific	3	5
1.5 Elective IGeneric/ Discipline Specific	3	4	2.5 Elective II Generic/ Discipline Specific	3	4	3.5 Elective III Generic/ Discipline Specific	3	4	4.5 Elective IV Generic/ Disciplin	3	3	5.5 Elective V Generic / Discipli	3	4	6.5 Elective VIII Generic/ Discipli	3	5

									e Specific			ne Specific			ne Specific		
1.6 Skill Enhancement Course SEC-1	2	2	2.6 Skill Enhancement Course SEC-2	2	2	3.6 Skill Enhancement Course SEC-4, (Entrepreneurial Skill)	1	1	4.6 Skill Enhancement Course SEC-6	2	2	5.6 Elective VI Generic / Discipline Specific	3	4	6.6 Extension Activity	1	-
1.7 Skill Enhancement - (Foundation Course)	2	2	2.7 Skill Enhancement Course –SEC-3	2	2	3.7 Skill Enhancement Course SEC-5	2	2	4.7 Skill Enhancement Course SEC-7	2	2	5.7 Value Education	2	2	6.7 Professional Competency Skill	2	2
						3.8 E.V.S.	-	1	4.8 E.V.S	2	1	5.8 Summer Internship /Industrial Training	2				
	23	**30**		**23**	**30**		**22**	**30**		**25**	**30**		**26**	**30**		**21**	**30**
Total – 140 Credits																	

B.Sc. Computer Science (CloudComputing andCyberSecurity) Curriculum (Subject/StructureofCourseStudy)

First year -SEMESTER - I

CODE	Subject	Hours	Credits	Internal Marks	External Marks
T1	Tamil	6	3	25	75
E1	English	6	3	25	75
Part-III	CC1 ProgramminginC	5	5	25	75
	CC2-Lab1:ProgramminginC Lab	5	5	25	75
	EC1-Discrete Mathematical Structures	4	3	25	75
Part IV	Skill Enhancement Course-SEC-1 (Non Major Elective)-Office Automation	2	2	25	75
	Foundation Course FC - Problem Solving Techniques	2	2	25	75
	Total	**30**	**23**		

SEMESTER - II

CODE	Subject	Hours	Credits	Internal Marks	External Marks
T2	Tamil	6	3	25	75
E2	English	6	3	25	75
Part-III	CC3-JavaProgramming	5	5	25	75
	CC4-Lab3:JavaProgramming Lab	5	5	25	75
	EC2- Numerical Methods	4	3	25	75
Part IV	Skill Enhancement Course- SEC-2 (Non Major Elective) - Quantitative Aptitude	2	2	25	75
	Skill Enhancement Course – SEC-3 (Discipline / Subject Specific) – Advanced Excel	2	2	25	75
	Total	**30**	**23**		

Second Year - SEMESTER - III

CODE	Subject	Hours	Credits	Internal Marks	External Marks
T3	Tamil	6	3	25	75
E3	English	6	3	25	75
Part III	CC5-DataStructures andComputer Algorithms	5	5	25	75
	CC6-Lab5:DataStructuresAnd ComputerAlgorithms Lab	5	5	25	75
	EC3-Probability and Statistics	4	3	25	75
Part IV	Skill Enhancement Course - SEC-4 (Entrepreneurial Based) – Python programming Lab	1	1	25	75
	Skill Enhancement Course - SEC-5 (Discipline Specific/ Generic) Digital Computer Fundamentals	2	2	25	75
	Environmental Studies	1	-	-	-
	Total	**30**	**22**		

SEMESTER - IV

CODE	Subject	Hours	Credits	Internal Marks	ExternalMarks
T4	Tamil	6	3	25	75
E4	English	6	3	25	75
PartIII	CC7-Cloud Computing	5	5	25	75
	CC8- CloudComputing Lab	5	5	25	75
	EC4 -Resource ManagementTechniques	3	3	25	75
ParIV	Skill Enhancement Course – SEC-6 - PHP Programming Web Application Development Lab	2	2	25	75
	Skill Enhancement Course - SEC-7- DataWarehousingandDataMining	2	2	25	75
	EVS	1	2	25	75
	Total	**30**	**25**		

Third Year SEMESTER – V

CODE	Subject	Hours	Credits	Internal Marks	External Marks
Part III	CC9-SoftwareEngineering	5	4	25	75
	CC10-Ethical hacking	5	4	25	75
	CC11-Network security	5	4	25	75
	EC5-BiometricSystems	4	3	25	75
	EC6-Lab 9: Cryptographyand NetworkSecurity lab	4	3	25	75
	CC12-Core/Project with viva voce	5	4	25	75
Part IV	Value Education	2	2	25	75
	Internship / Industrial Training (Summer vacation at the end of IV semester activity)	-	2	-	-
	Total	**30**	**26**		

SEMESTER - VI

CODE	Subject	Hours	Credits	Internal Marks	External Marks
	CC13- Internet of Things	6	4	25	75
	CC14-Machine Learning	6	4	25	75
	CC15-Lab11:CyberSecurity Lab	6	4	25	75
	EC7- Big Data Analytics	5	3	25	75
	EC8- Information Security	5	3	25	75
Part IV	Professional Competency Skill Enhancement Course SEC8-Quantitative Aptitude	2	2	25	75
Part V	Extension Activity	-	1	-	-
	Total	**30**	**21**		

SEMESTER I

CourseCode: CC1	**PROGRAMMING IN C**		**Credits 5**
LectureHours:(L) 5 perweek	**TutorialHours: (T)perweek**	**LabPractice Hours: (P)perweek**	**Total:(L+T+P) perweek 5**
CourseCategory: Core	**Year&Semester: Year I Semester I**	**AdmissionYear: 2023-2024**	
Pre-requisite :			

LearningObjectives:

- To gain knowledge in C language.
- To inculcate fundamental programming skills.

CourseOutcomes:(forstudents:Toknowwhattheyaregoingtolearn)

CO1: Remember the program structure of C with its syntax and semantics

CO2: Understand the programming principles in C (data types, operators, branching and looping, arraysfunctions, structures, pointers and files)

CO3: Apply the programming principles learnt in real-time problems

CO4: Analyze the various methods of solving a problem and choose the best method

CO5: Code, debug and test the programs with appropriate test cases

Recap:(notforexamination)Motivation/previouslecture/relevantportionsrequiredforthe course)[Thisisdoneduring2Tutorialhours)

Units	Contents	RequiredHours
I	Overview of C: Importance of C, sample C program, C program structure, executing C program. Constants, Variables, and Data Types: Character set, C tokens, keywords and identifiers, constants, variables, data types, declaration of variables, Assigning values to variables---Assignment statement, declaring a variable as constant, as volatile. Operators and Expression: Arithmetic, Relational, logical, assignment, increment, decrement, conditional, bitwise and special operators, arithmetic expressions, operator precedence, type conversions, mathematical functions Managing Input and Output Operators: Reading and writing a character, formatted input, formatted output.	**15**
II	Decision Making and Branching: Decision making with If, simple IF, IF ELSE, nested IF ELSE , ELSE IF ladder, switch, GOTO statement. Decision Making and Looping: While, Do-While, For, Jumps in loops.	**15**
III	Arrays: Declaration and accessing of one & two-dimensional arrays, initializing two-dimensional arrays, multidimensional arrays. Functions: The form of C functions, Return values and types, calling a function, categories of functions, Nested functions,	**15**

	Recursion, functions with arrays, call by value, call by reference, storage classes-character arrays and string functions.	
IV	Structures and Unions: Defining, giving values to members, initialization and comparison of structure variables, arrays of structure, arrays within structures, structures within structures, structures and functions, unions. Preprocessors: Macro substitution, file inclusion.	**15**
V	Pointers: definition, declaring and initializing pointers, accessing a variable through address and through pointer, pointer expressions, pointer increments and scale factor, pointers and arrays, pointers and functions, pointers and structures. File Management in C: Opening, closing and I/O operations on files, random access to files, command line arguments.	**15**
ExtendedProfessionalComponent(isapartofinternalcomponent .	Questionsrelatedtotheabovetopics,fromvariouscompetitiveexaminationsUPSC/TRB/NET/UGC–CSIR/GATE/TNPSC/otherstobesolved(Tobediscussedduringthe Tutorialhour)	
Skillsacquiredfrom the course	Knowledge,ProblemSolving,Analyticalability,ProfessionalCompetency,ProfessionalCommunicationandTransferrable Skill	

LearningResources:

- **RecommendedTexts**
  1. E. Balagurusamy, Programming in ANSI C, Fifth Edition, Tata McGraw-Hill, 2010.
- **ReferenceBooks**
  1. Byron Gottfried, Schaum‟s Outline Programming with C, Fourth Edition, Tata McGraw-Hill, 2018.
  2. Kernighan and Ritchie, The C Programming Language, Second Edition, Prentice Hall, 1998.
  3. YashavantKanetkar, Let Us C, Eighteenth Edition, BPB Publications,2021
- **Webresources :** Web resources from NDL Library, E-content from open-source libraries

Course Outcomes	
Course Outcomes	**On completion of this course, students will be able;**
CO1	**Remember the program structure of C with its syntax and semantics**
CO2	**Understand the programming principles in C (data types, operators, branching and looping, arrays, functions, structures, pointers and files)**
CO3	**Apply the programming principles learnt in real-time problems**
CO4	**Analyze the various methods of solving a problem and choose the best method**
CO5	**Code, debug and test the programs with appropriate test cases**

Mapping with Programme Outcomes:

	PO 1	**PO 2**	**PO 3**	**PO 4**	**PO 5**	**PO 6**
CO1	**S**	**S**	**S**	**S**	**S**	**S**
CO2	**S**	**S**	**S**	**S**	**S**	**S**
CO3	**M**	**M**	**S**	**M**	**M**	**S**
CO4	**S**	**S**	**S**	**S**	**S**	**M**
CO5	**S**	**S**	**S**	**S**	**M**	**S**

3 – Strong, 2 – Medium , 1 - Low

<table>
<tr><td colspan="2">CourseCode : CC2</td><td colspan="2">PROGRAMMING IN C LAB</td><td>Credits 5</td></tr>
<tr><td colspan="2">LectureHours:(L)</td><td>Tutorial Hours: (T)perweek</td><td>LabPractice Hours: (P)perweek 5</td><td>Total:(L+T+P) perweek 5</td></tr>
<tr><td colspan="2">CourseCategory: Core</td><td colspan="2">Year&Semester: Year I Semester I</td><td>AdmissionYear: 2023-2024</td></tr>
<tr><td colspan="2">Pre-requisite</td><td colspan="3">Basic knowledge in programming in c</td></tr>
<tr><td colspan="5">LearningObjectives:
• To implement programming skills using C
• To impart knowledge and provide efficient solutions for real time problems using C language</td></tr>
<tr><td colspan="5">CourseOutcomes:(forstudents:Toknowwhattheyaregoingtolearn)
CO1:Remember and understand how to write programs using the basic syntax and semantics in
CO2:Apply the concepts of functions, macros, arrays, structures, pointers and files in programs to solve problems
CO3:Analyze and understand programs written in C language
CO4: Evaluate the program execution flow with test cases and apply debugging
CO5: Design algorithms and write programs in C language for the given problems</td></tr>
<tr><td colspan="5">Recap:(notforexamination)Motivation/previouslecture/relevantportionsrequiredforthe course)[Thisisdoneduring2Tutorialhours)</td></tr>
<tr><td>Units</td><td colspan="3">Contents</td><td>RequiredHours</td></tr>
<tr><td>I</td><td colspan="3">Variables, Data types, Constants and Operators
1. Evaluation of expression ex: ((x+y) ^2 * (x+z))/w
2. Temperature conversion problem (Fahrenheit to Celsius)
3. Program to convert days to months and days (Ex: 364 days = 12 months and 4 days)
4. Solution of quadratic equation
5. Salesman salary (Given: Basic Salary, Bonus for every item sold, commission on the total monthly sales)</td><td>75</td></tr>
<tr><td>II</td><td colspan="3">Decision making Statements
6. Maximum of three numbers
7. Calculate Square root of five numbers (using goto statement)
8. Pay-Bill Calculation for different levels of employee (Switch statement)
9. Fibonacci series
10.Floyds Triangle
11.Pascal‟s Triangle</td><td></td></tr>
</table>

III	**Arrays, Functions and Strings** 12. Prime numbers in an array 13. Sorting data (Ascending and Descending) 14. Matrix Addition and Subtraction 15. Matrix Multiplication 16. Function with no arguments and no return values	

	17. Function that convert lower case letters to upper case 18. Factorial using recursion. 19. Perform String Operations using Switch Case.	
IV	**Structures and Macros** 20. Structure that describes a Hotel (name, address, grade, avg room rent, number of rooms) Perform some operations (list of hotels of a given grade etc.) 21. Using Pointers in Structures. 22. Cricket team details using Union. 23. Write a macro that calculates the max and min of two numbers 24. Nested macro to calculate Cube of a number.	
V	**Pointers and Files** 25. Evaluation of Pointer expressions 26. Function to exchange two pointer values 27. Creation, insertion and deletion in a linked list 28. Program to read a file and print the data. 29. Program to receive a file name and a line of text as command line arguments and write the text to the file 30. Program to copy the content of one file to anotherfile.	
ExtendedProfessionalComponent(is apartofinternalcomponent only,Notto be included in the ExternalExaminationquestion paper)	Questionsrelatedtotheabovetopics,fromvariouscompetitiveexaminationsUPSC/TRB/NET/UGC–CSIR/GATE/TNPSC/otherstobesolved(Tobediscussedduringthe Tutorialhour)	
Skillsacquiredfrom the course	Knowledge,ProblemSolving,Analyticalability,Professional Competency,ProfessionalCommunicationandTransferrable Skill	

LearningResources:

- **RecommendedTexts**
  1. E. Balagurusamy, Programming in ANSI C, Fifth Edition, Tata McGraw-Hill, 2010.
- **ReferenceBooks**
  1. Byron Gottfried, Schaum‟s Outline Programming with C, Fourth Edition, Tata McGraw-Hill, 2018.
  2. Kernighan and Ritchie, The C Programming Language, Second Edition, Prentice

Hall, 1998.

3. YashavantKanetkar, Let Us C, Eighteenth Edition, BPB Publications, 2021

- **Webresources:** Web resources from NDL Library, E-content from open-source libraries

Course Outcomes	Course Outcomes
	On completion of this course, students will be able;
CO1	Remember and understand how to write programs using the basic syntax and semantics in C
CO2	Apply the concepts of functions, macros, arrays, structures, pointers and files in programs to solve problems
CO3	Analyze and understand programs written in C language
CO4	Evaluate the program execution flow with test cases and apply debugging
CO5	Design algorithms and write programs in C language for the given problems.

Mapping with Programme Outcomes :

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6
CO1	**S**	**S**	**S**	**S**	**S**	**S**
CO2	**S**	**S**	**S**	**S**	**M**	**S**
CO3	**S**	**S**	**S**	**M**	**S**	**S**
CO4	**S**	**M**	**S**	**M**	**S**	**M**
CO5	**S**	**S**	**S**	**S**	**M**	**S**

3 – Strong, 2 – Medium , 1 - Low

CourseCode: SEC1	**OFFICE AUTOMATION**		**Credits: 2**
LectureHours:(L) perweek: 2	**TutorialHours: (T)perweek**	**LabPractice Hours: (P)perweek**	**Total:(L+T+P) perweek: 2**
CourseCategory: SEC1	**Year&Semester: I Year I Semester**		**AdmissionYear: 2023-2024**
Pre-requisite	Basic skills in Computer operations		

LearningObjectives:(forteachers:whattheyhavetodointheclass/lab/field)

- The major objective in introducing the Computer Skills course is to impart training forstudents in Microsoft Office which has different components like MS Word, MS Excel andPowerpoint.
- Thecourseishighlypracticeorientedratherthanregularclassroomteaching.
- Toacquireknowledgeoneditor,spreadsheetandpresentationsoftware.

CourseOutcomes:(forstudents:Toknowwhattheyaregoingtolearn)

CO1: Understandthebasicsofcomputersystemsanditscomponents.

CO2: Understand and applythebasicconceptsofawordprocessingpackage.

CO3: Understandand applythebasicconceptsofelectronicspreadsheetsoftware.

CO4: Understandandapplythebasicconceptsofdatabasemanagementsystem.

CO5: UnderstandandcreateapresentationusingPowerPointtool.

Recap:(notforexamination)Motivation/previouslecture/relevantportionsrequiredforthe course)[Thisisdoneduring2Tutorialhours)

Units	Contents	RequiredHours
I	**Introductory concepts:** Memory unit– CPU-Input Devices: Key board, Mouse and Scanner.Outputdevices:Monitor,Printer.Introductionto Operatingsystems&itsfeatures:DOS– UNIX– Windows. IntroductiontoProgrammingLanguages.	6
II	**Word Processing:** Open, Save and close word document; Editing text – tools, formatting, bullets;SpellChecker - Document formatting – Paragraph alignment, indentation, headers and footers,numbering;printing–Preview,options,merge.	6
III	**Spreadsheets:**Excel– opening,enteringtextanddata,formatting,navigating;For mulas–entering,handlingand copying;Charts– creating,formatting and printing,analysistables,preparationoffinancialstatement s,introductiontodataanalytics.	6
IV	**Database Concepts:** The concept of data base management system; Data field, records, and files,Sorting and indexing data; Searching records. Designing queries, and reports; Linking of datafiles; Understanding Programming environment in DBMS;	6

	Developing menu drive applicationsinquerylanguage(MS–Access).	
V	**Power point:** Introduction to Power point - Features – Understanding slide typecasting & viewingslides – creating slide shows. Applying special object – including objects & pictures – Slidetransition– Animationeffects,audioinclusion,timers.	**6**
ExtendedProfessional Component(isapartofinternalcomponent only,Nottobe included in the ExternalExaminationquestion paper)	Questionsrelatedtotheabovetopics,fromvariouscompetitiveexaminationsUPSC/TRB/NET/UGC–CSIR/GATE/TNPSC/otherstobesolved(Tobe discussedduringtheTutorialhour)	
Skillsacquiredfrom the course	Knowledge,ProblemSolving,Analyticalability,Professional Competency,ProfessionalCommunicationandTransferrable Skill	

LearningResources:

- **RecommendedTexts**

1. PeterNorton,"IntroductiontoComputers"–TataMcGraw-Hill.

- **ReferenceBooks**

1. JenniferAckermanKettel,GuyHat-Davis,CurtSimmons,"Microsoft2003",TataMcGraw-Hill.

- **Webresources :** Web content from NDL / SWAYAM or open source web resources

Mapping with Programme Outcomes:

	PO 1	**PO 2**	**PO 3**	**PO 4**	**PO 5**
CO1	**3**	**3**	**3**	**2**	**2**
CO2	**3**	**3**	**3**	**3**	**1**
CO3	**3**	**3**	**3**	**2**	**1**
CO4	**3**	**3**	**2**	**3**	**1**
CO5	**3**	**3**	**3**	**2**	**3**

3 – Strong, 2 – Medium , 1 - Low

CourseCode: FC	**PROBLEM SOLVING TECHNIQUES**		**Credits: 2**
LectureHours:(L) perweek: 2	**TutorialHours: (T)perweek**	**LabPractice Hours: (P)perweek**	**Total:(L+T+P) perweek: 2**
CourseCategory: FC	**Year&Semester: I Year I Semester**		**AdmissionYear: 2023-2024**
Pre-requisite	Basic of Problem-solving skills		

LearningObjectives:

- To understand the importance of algorithms and programs, and to know of the basic problem solving strategies.
- To learn efficient strategies and algorithms to solve standard problems, thus laying a firm foundation for designing algorithmic solutions to problems.

CourseOutcomes:(forstudents:Toknowwhattheyaregoingtolearn)

CO1: Understand the systematic approach to problem solving.

CO2: Know the approach and algorithms to solve specific fundamental problems.

CO3: Understand the efficient approach to solve specific factoring-related problems.

CO4: Understand the efficient array-related techniques to solve specific problems.

CO5: Understand the efficient methods to solve specific problems related to text processing. Understand how recursion works.

Recap:(notforexamination)Motivation/previouslecture/relevantportionsrequiredforthe course)[Thisisdoneduring2Tutorialhours)

Units	Contents	RequiredHours
I	**Introduction:** Notion of algorithms and programs – Requirements for solving problems by computer – The problem-solving aspect: Problem definition phase, Getting started on a problem, The use of specific examples, Similarities among problems, Working backwards from the solution – General problem-solving strategies - Problem solving using top-down design – Implementation of algorithms – The concept of Recursion.	**6**
II	**Fundamental Algorithms**: Exchanging the values of two variables – Counting - Summation of a set of numbers - Factorial computation - Sine function computation - Fibonacci Series generation - Reversing the digits of an integer – Base Conversion.	**6**

III	**Factoring Methods**: Finding the square root of a number – The smallest divisor of an integer – Greatest common divisor of two integers - Generating prime numbers – Computing the prime factors of an integer – Generation of pseudo-random numbers - Raising a number to a large power – Computing the *n*th Fibonacci number.	**6**
IV	**Array Techniques**: Array order reversal – Array counting or histogramming – Finding the maximum number in a set - Removal of duplicates from an ordered array - Partitioning an array – Finding the k^{th} smallest element – Longest monotone subsequence.	**6**
V	**Text Processing and Pattern Searching**: Text line length adjustment – Left and right justification of text – Keyword searching in text – Text line editing – Linear pattern search. **Recursive algorithms**: Towers of Hanoi – Permutation generation.	**6**

Course Outcomes	
Course Outcomes	On completion of this course, students will be able;
CO1	Understanding basic systematic approach to problem solving.
CO2	Learn the approach and algorithms to solve specific fundamental problems.
CO3	Studying the efficient approach to solve specific factoring-related problems.
CO4	To know the efficient array-related techniques to solve specific problems.
CO5	Understand the efficient methods to solve specific problems related to text processing and how recursion works.

ExtendedProfessionalComponent(is apartofinternalcomponent only,Nottobe included in the ExternalExaminationquestion paper)	Questionsrelatedtotheabovetopics,fromvariouscompetitive examinationsUPSC/TRB/NET/UGC–CSIR/GATE/TNPSC/otherstobesolved(TobediscussedduringtheTutorialhour)	

Skillsacquiredfrom the Course	Knowledge,ProblemSolving,Analyticalability,Professional Competency,ProfessionalCommunicationandTransferrable Skill	

LearningResources:

- **RecommendedTexts**

1. R. G. Dromey, *How to Solve it by Computer*, Pearson India, 2007.

- **ReferenceBooks**

1. George Polya, Jeremy Kilpatrick, *The Stanford Mathematics Problem Book: With Hints and Solutions*, Dover Publications, 2009 (Kindle Edition 2013).
2. Greg W. Scragg, *Problem Solving with Computers*, Jones & Bartlett 1st edition, 1996.

- **Webresources:** Web resources from NDL Library, E-content from open-source libraries

Mapping with Programme Outcomes:

	PO 1	**PO 2**	**PO 3**	**PO 4**	**PO 5**	**PO 6**
CO1	M	L	M	S	M	S
CO2	S	L	M	L	S	L
CO3	L	S	L	M	S	M
CO4	M	S	S	L	S	M
CO5	S	M	M	S	L	S

3 – Strong, 2 – Medium , 1 - Low

SEMESTER II

CourseCode: CC3	**JAVA PROGRAMMING**		**Credits: 5**
LectureHours:(L) perweek: 5	**TutorialHours: (T)perweek**	**LabPractice Hours: (P)perweek**	**Total:(L+T+P) perweek: 5**
CourseCategory:CC3	**Year&Semester: I year & II semester**		**AdmissionYear: 2023-2024**
Pre-requisite	Basic Programming skill		

LearningObjectives:(forteachers:whattheyhavetodointheclass/lab/field)

- To provide fundamental knowledge of object-oriented programming.
- To equip the student with programming knowledge in Core Java from the basics up.
- To enable the students to use AWT controls, Event Handling and Swing for GUI.

CourseOutcomes:(forstudents:Toknowwhattheyaregoingtolearn)

CO1: Understand the basic Object-oriented concepts.

Implement the basic constructs of Core Java

CO2: Implement inheritance, packages, interfaces and exception handling of Core Java.

CO3: Implement multi-threading and I/O Streams of Core Java

CO4: Implement AWT and Event handling.

CO5: Use Swing to create GUI.

Recap:(notforexamination)Motivation/previouslecture/relevantportionsrequiredforthe course)[Thisisdoneduring2Tutorialhours)

Units	Contents	RequiredHours
I	**Introduction:**ReviewofObject Orientedconcepts - HistoryofJava - Javabuzzwords - JVMarchitecture - Datatypes - Variables - Scope and life timeofvariables - arrays - operators - controlstatements - type conversion and casting - simple java program - constructors - methods - Static block - Static Data - StaticMethodStringand StringBufferClasses	**15**
II	**Inheritance**: Basic concepts - Types of inheritance - Member access rules - Usage of this and Super key word - Method Overloading - Method overriding - Abstract classes - Dynamic method dispatch - Usage of final keyword. **Packages**:Definition-AccessProtection - ImportingPackages. **Interfaces**:Definition–Implementation–Extending Interfaces. **Exception Handling**: *try – catch - throw - throws – finally* – Built-inexceptions - Creating own Exception classes.	**15**
III	**Multithreaded Programming**: Thread Class - Runnable interface –Synchronization–Usingsynchronizedmethods–	**15**

	Using*synchronized*statement- InterthreadCommunication –Deadlock. **I/O Streams:** Concepts of streams - Stream classes- Byte and Character stream - Reading console Input and Writing Console output - File Handling.	
IV	**AWT Controls:** The AWT class hierarchy - user interface components- Labels - Button - Text Components - Check Box - Check Box Group - Choice - List Box - Panels – Scroll Pane - Menu - Scroll Bar. Working with Frame class - Colour - Fonts and layout managers. **Event Handling:** Events - Event sources - Event Listeners - Event Delegation Model (EDM) - Handling Mouse and Keyboard Events - Adapter classes - Inner classes.	**15**
V	**Swing:** Introduction to Swing - Hierarchy of swing components. Containers - Top level containers - JFrame - JWindow - JDialog - JPanel - JButton - JToggleButton - JCheckBox - JRadioButton - JLabel,JTextField - JTextArea - JList - JComboBox – JscrollPane	**15**
ExtendedProfessionalComponent(is apartofinternalcomponent .	Questionsrelatedtotheabovetopics,fromvariouscompetitive examinationsUPSC/TRB/NET/UGC–CSIR/GATE/TNPSC/otherstobesolved(TobediscussedduringtheTutorialhour)	
Skillsacquiredfrom the course	Knowledge,ProblemSolving,Analyticalability,Professional Competency,ProfessionalCommunicationandTransferrable Skill	

LearningResources:

- **RecommendedTexts**
  1. Herbert Schildt, The Complete Reference, Tata McGraw Hill, New Delhi, 7th Edition, 2010.
  2. Gary Cornell, Core Java 2 Volume I – Fundamentals, Addison Wesley, 1999.
- **ReferenceBooks**

1. Head First Java, O"Rielly Publications,
2. Y. Daniel Liang, *Introduction to Java Programming*, 7th Edition, Pearson Education India, 2010.

Webresources: Web resources from NDL Library, E-content from open-source libraries

Mapping with Programme Outcomes:

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6
CO1	L	M	S	M	M	S
CO2	S	L	S	M	S	L
CO3	M	S	L	M	M	S
CO4	L	S	S	L	S	M
CO5	S	M	M	S	L	S

3 – Strong, 2 – Medium , 1 – Low

CourseCode: CC4	**JAVA PROGRAMMING LAB**		**Credits: 5**
LectureHours:(L)	**TutorialHours: (T)perweek**	**LabPractice Hours: (P)perweek: 5**	**Total:(L+T+P) perweek: 5**
CourseCategory:Practical	**Year&Semester: I Year & II Semester**		**AdmissionYear: 2023-2024**
Pre-requisite	Basic Programming debugging skills		
LearningObjectives:(forteachers:whattheyhavetodointheclass/lab/field) • To gain practical expertise in coding Core Java programs • To become proficient in the use of AWT, Event Handling and Swing.			
Recap:(notforexamination)Motivation/previouslecture/relevantportionsrequiredforthe			

	List of Exercises:	**RequiredHours**

	1. Write a Java program that prompts the user for an integer and then prints out all the prime numbers up to that Integer? 2. Write a Java program to multiply two given matrices. 3. Write a Java program that displays the number of characters, lines and words in a text? 4. Generaterandomnumbersbetweentwogivenlimitsusing Random classandprintmessagesaccordingtotherangeofthevaluegenerated. 5. WriteaprogramtodoStringManipulationusingCharacterArrayand performthefollowingstringoperations: a. Stringlength b. Findingacharacterataparticularposition c. Concatenatingtwostrings 6. WriteaprogramtoperformthefollowingstringoperationsusingString class: a. StringConcatenation b. Searchasubstring c. Toextractsubstringfromgivenstring 7. Writeaprogram toperform stringoperationsusingStringBufferclass: a. Lengthof astring b. Reverseastring c. Deleteasubstringfrom thegiven string 8. Write a java program that implements a multi-thread	**75**

	application that has three threads. First thread generates random integer every 1 second and if the value is even, second thread computes the square of the number and prints. If the value is odd, the third thread will print the value of cube of the number. 9. Writeathreadingprogramwhichusesthesamemethodasynchronouslytoprintthenumbers1to10usingThread1andtoprint90to100using Thread2. 10. Writeaprogram to demonstrate the useoffollowingexceptions. a. ArithmeticException b. NumberFormatException 11. Write a Java program that reads on file name from the user, then displays information about whether the file exists, whether the file is readable, whether the file is writable, the type of file and the length of the file in bytes? 12. Writeaprogramtoacceptatextandchangeitssizeandfont.Include bolditalicoptions.Useframesandcontrols. 13. Write a Java program that handles all mouse events and shows the event name at the center of the window when a mouse event is fired. (Use adapter classes).	

LearningResources:

LearningResources:

- **RecommendedTexts**
  3. Herbert Schildt, The Complete Reference, Tata McGraw Hill, New Delhi, 7th Edition, 2010.
  4. Gary Cornell, Core Java 2 Volume I – Fundamentals, Addison Wesley, 1999.
- **ReferenceBooks**
  3. Head First Java, O"Rielly Publications,

Webresources: Web resources from NDL Library, E-content from open-source libraries

Mapping with Programme Outcomes:

Course Outcomes	
Course Outcomes	On completion of this course, students will be able;
CO1	Code, debug and execute the Java problems
CO2	Implement multi-threading and exception-handling.
CO3	Implement File Handling.
CO4	Demonstrate Event Handling and Implement Synchronous and Asynchronous programming.
CO5	Create GUI using Swing and AWT and apply event handling.

	PO 1	**PO 2**	**PO 3**	**PO 4**	**PO 5**	**PO 6**
CO1	L	M	S	M	L	S
CO2	S	S	L	M	L	M
CO3	M	L	M	L	M	L
CO4	L	L	S	M	L	S
CO5	M	M	S	S	L	M

3 – Strong, 2 – Medium , 1 – Low

CourseCode: SEC3	**ADVANCED EXCEL**		**Credits: 2**
LectureHours:(L) perweek: 2	**TutorialHours: (T)perweek**	**LabPractice Hours: (P)perweek**	**Total:(L+T+P) perweek: 2**
CourseCategory: Skill Enhancement Course	**Year&Semester: I Year II Semester**		**AdmissionYear: 2023-2024**
Pre-requisite	Basic knowledge in office automation / Excel		

LearningObjectives:(forteachers:whattheyhavetodointheclass/lab/field)

The objective of this course is to help the students learn the advanced features of Excel, to summarise, analyse, explore, and present visualisations of data in the form of charts, graphs.

CourseOutcomes:(forstudents:Toknowwhattheyaregoingtolearn)

CO1: Handle large amounts of data

CO2: Aggregate numeric data and summarise into categories and subcategories

CO3: Filtering, sorting, and grouping data or subsets of data

CO4: Create pivot tables to consolidate data from multiple files

CO5: Presenting data in the form of charts and graphs

Recap:(notforexamination)Motivation/previouslecture/relevantportionsrequiredforthe course)[Thisisdoneduring2Tutorialhours)

Units	**Contents**	**RequiredHours**

I	Basics of Excel- Customizing common options- Absolute and relative cells- Protecting and un-protecting worksheets and cells- Working with Functions - Writing conditional expressions - logical functions - lookup and reference functions- VlookUP with Exact Match, Approximate Match- Nested VlookUP with Exact Match- VlookUP with Tables, Dynamic Ranges- Nested VlookUP with Exact Match- Using VLookUP to consolidate Data from Multiple Sheets	**6**

II	Data Validations - Specifying a valid range of values - Specifying a list of valid values- Specifying custom validations based on formula - Working with Templates Designing the structure of a template- templates for standardization of worksheets - Sorting and Filtering Data -Sorting tables- multiple-level sorting- custom sorting- Filtering data for selected view - advanced filter options- Working with Reports Creating subtotals- Multiple-level subtotal.	**6**
III	Creating Pivot tables Formatting and customizing Pivot tables- advanced options of Pivot tables- Pivot charts- Consolidating data from multiple sheets and files using Pivot tables- external data sources- data consolidation feature to consolidate data- Show Value As % of Row, % of Column, Running Total, Compare with Specific Field- Viewing Subtotal under Pivot- Creating Slicers.	**6**
IV	More Functions Date and time functions- Text functions- Database functions- Power Functions - Formatting Using auto formatting option for worksheets- Using conditional formatting option for rows, columns and cells- WhatIf Analysis - Goal Seek- Data Tables- Scenario Manager.	**6**
V	Charts - Formatting Charts- 3D Graphs- Bar and Line Chart together- Secondary Axis in Graphs- Sharing Charts with PowerPoint / MS Word, Dynamically- New Features Of Excel Sparklines, Inline Charts, data Charts- Overview of all the new features.	**6**
ExtendedProfessionalComponent(isapartofinternalcomponent.	Questionsrelatedtotheabovetopics,fromvariouscompetitive examinationsUPSC/TRB/NET/UGC–CSIR/GATE/TNPSC/otherstobesolved(TobediscussedduringtheTutorialhour)	

Skillsacquiredfrom the course	Knowledge,ProblemSolving,Analyticalability,Professional Competency,ProfessionalCommunicationandTransferrable Skill	

LearningResources:

- **RecommendedText**

  Excel 2019 All-in-One For Dummies – 2018- Greg Harvey

- **ReferenceBooks**

  Microsoft Excel 2019 Pivot Table Data Crunching-2019,Bill Jelen and Michael Alexander

- **Webresources:** Web resources from NDL Library, E-content from open source libraries

Course Outcomes	
Course Outcomes	On completion of this course, students will be able;
CO1	Basics of excel, Multiple Spread sheets
CO2	Implement Data Validation , Templates
CO3	Implement Pivot Tables
CO4	Implement functions & Tables
CO5	Implement of Charts, Graphs, features of excel.

Mapping with Programme Outcomes:

	PO 1	**PO 2**	**PO 3**	**PO 4**	**PO 5**
CO1	3	3	3	2	2
CO2	3	3	3	3	1
CO3	3	3	3	2	1
CO4	3	3	2	3	1
CO5	3	3	3	2	3

3 – Strong, 2 – Medium , 1 - Low