Anekant Education Society's

Tuljaram Chaturchand College of Arts, Science and Commerce, Baramati (Autonomous)

B.Sc. (Computer Science) Academic Year 2022 – 2023

- F.Y.B.Sc.(Computer Science) Credit Structure -

Subject	**Semester – I**				**Semester – II**				**Total**
	Paper		**Practical Paper**		**Paper**		**Practical Paper**		
	I	**II**	**III**	**IV**	**I**	**II**	**III**	**IV**	
Computer Science	2	2	2	2	2	2	2	2	16
Mathematics	2	2	2	--	2	2	2	--	12
Electronics	2	2	2	--	2	2	2	--	12
Statistics	2	2	2	--	2	2	2	--	12
Physical Education	--	--	--	--	--	--	--	--	2
Democracy, Election & Governance	--	--	--	--	--	--	--	--	2

Total Number of Credits= 52(Core) + 2(Physical Education) + 2(Democracy, Election and Governance) = 56
1 Theory Credit = 18 Lectures

- S.Y.B.Sc.(Computer Science) Credit Structure –

Subject	**Semester – I**				**Semester – II**				**Total**
	Paper		**Practical Paper**		**Paper**		**Practical Paper**		
	I	**II**	**III**	**IV**	**I**	**II**	**III**	**IV**	
Computer Science	3	3	2	2	3	3	2	2	20
Mathematics	3	3	2	--	3	3	2	--	16
Electronics	3	3	2	--	3	3	2	--	16
English	3	--	--	--	3	--	--	--	6
Environment Awareness	--	--	--	--	--	--	--	--	2
Certificate Courses	--	--	--	--	--	--	--	--	2

Total Number of Credits= 58(Core) + 2(Env.Awareness) + 2(Certificate Course) = 62
1 Theory Credit= 16 Lectures

- T.Y.B.Sc.(Computer Science) Credit Structure –
Semester V

Subject	Theory						Practical			Total
	I	II	III	IV	V	VI	Lab Course I	Lab Course II	Lab Course III	
Computer Science	**3**	**3**	**3**	**3**	**3**	**3**	**2**	**2**	**2**	**24**

- T.Y.B.Sc.(Computer Science) Credit Structure –
Semester VI

Subject	Theory						Practical			Total
	I	II	III	IV	V	VI	Lab Course I	Lab Course II	Lab Course III	
Computer Science	**3**	**3**	**3**	**3**	**3**	**3**	**2**	**2**	**2**	**24**

Total Number of Credits= 48(Core)

1 Theory Credit = 16 Lectures

B.Sc.(Computer Science): Total Credits = 56 + 62 + 48 = 166

Proposed Syllabus structure under 2022 pattern

F.Y.B.Sc.(C.S.) Semester – I

Paper Code (2019 Pattern)	**Paper Title**	**Paper Code (Proposed 2022 Pattern)**	**Paper Title**
CSCO 1101	Basic Programming using C	UCSCO111	Basic Programming using C
CSCO 1102	DBMS-I	UCSCO112	DBMS-I
CSCO 1103	Lab. Course I : Basic programming using C	UCSCO113	Lab. Course I : Basic programming using C
CSCO 1104	Lab. Course II : DBMS I	UCSCO114	Lab. Course II : DBMS I

F.Y.B.Sc.(C.S.) Semester – II

Paper Code (2019 Pattern)	**Paper Title**	**Paper Code (Proposed 2022 Pattern)**	**Paper Title**
CSCO 1201	Advanced Programming using C	UCSCO121	Advanced Programming using C
CSCO 1202	DBMS-II	UCSCO122	DBMS-II
CSCO 1203	Lab. Course I : Advanced Programming using C	UCSCO123	Lab. Course I : Advanced Programming using C
CSCO 1204	Lab. Course II : DBMS II	UCSCO124	Lab. Course II : DBMS II

Class : F.Y.B.Sc. (Computer Science)			
Semester I		**Semester II**	
UCSCO111	Basic Programming using C	UCSCO121	Advanced Programming using C
UCSCO112	DBMS-I	UCSCO122	DBMS-II
UCSCO113	Lab. Course I : Basic programming using C	UCSCO123	Lab. Course I : Advanced Programming using C
UCSCO114	Lab. Course II : DBMS I	UCSCO124	Lab. Course II : DBMS II
Physical Education		Democracy, Election & Governance	

Course Structure for F.Y.B.Sc.(Computer Science) (2022 Pattern)

Subject: **Computer Science**

Semester	Paper Code	Title of Paper	No. of Credits	Exam.	Marks
I	UCSCO111	Basic Programming using C	2	I/E	60+40
	UCSCO112	DBMS-I	2	I/E	60+40
	UCSCO113	Lab. Course I : Basic programming using C	2	I/E	60+40
	UCSCO114	Lab. Course II : DBMS I	2	I/E	60+40
II	UCSCO121	Advanced Programming using C	2	I/E	60+40
	UCSCO122	DBMS-II	2	I/E	60+40
	UCSCO123	Lab. Course I : Advanced Programming using C	2	I/E	60+40
	UCSCO124	Lab. Course II : DBMS II	2	I/E	60+40
		Physical Education	2	--	--
		Democracy, Election & Governance	2	--	--

Paper I: UCSCO111 Basic Programming Using C

Course Objectives:

Students successfully completing this course will be able:

1. To understand and design algorithm for problem solving
2. To develop Problem Solving abilities using computers
3. To develop algorithms, flowcharts for problems
4. To learn block structured and procedural programming language
5. To develop skills for writing programs using 'C'
6. To test, debug and execute programs.

Course Outcomes:

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

1. Explore algorithmic approaches to problem solving.
2. Develop programs using control structures and arrays in 'C'.

PAPER CODE : UCSCO111 **PAPER – I : Basic Programming Using C** **(Credits – 02 No. of Lectures – 36)**		
	Chapter and Sub Topics	**No. of Lectures**
Unit – I	**Problem-Solving Using Computer** 1.1 Problem Solving 1.2 Algorithms & Flowcharts 1.3 Programming Languages Machine language Assembly language High level languages **1.4 Converting pseudo-code to programs.** **1.5 Programming tools**	5
Unit – II	**Introduction to C** 2.1 History 2.2 Structure of a C program 2.3 Application Areas 2.4 C Program development life cycle 2.5 Sample programs	3
Unit – III	**C Tokens** 3.1 Keywords 3.2 Identifiers 3.3 Variables 3.4 Constants – character, integer, float, string, escape sequences 3.5 Data types – built-in and user defined 3.6 Operators and Expressions Operator types (arithmetic, relational, logical,assignment, bitwise, conditional, other operators), precedence and associativity rules.	7
Unit – IV	**Control Structures** 4.1 Decision making structures if, if-else, switch-case 4.2 Loop Control structures While, do-while, for 4.3 Jumping Statements break, continue and goto 4.4 Nested structures	7
Unit – V	**Functions in C** 5.1 What is a function	8

	5.2 Advantages of Functions 5.3 Standard library functions 5.4 User defined functions :Declaration, definition, function call, parameter passing (by value), return keyword 5.5 Scope of variables, storage classes 5.6 Recursion	
Unit – VI	**Arrays** 6.1 Array Concept 6.2 Types – one, two and multidimensional 6.3 Array Operations - declaration, Initialization, accessing array elements 6.4 Passing arrays to functions **6.5 Array Applications**	6

References:

1. Yashavant Kanetkar : Let Us C 7th Edition, PBP Publications
2. E Balaguruswamy : Programming in ANSI C 7th Edition, Tata Mc-Graw Hill Publishing Co.Ltd.-New Delhi
3. Brian W. Kernighan and Dennis M. Ritchie : The C Programming Language 2nd Edition, Prentice Hall Publication
4. The Complete Reference to C, Herbert Schildt
5. Problem Solving with C, Harrow
6. Programming in C ,A Practical Approach, Ajay Mittal , Pearson

Class : F.Y. B. Sc. (Computer Science) (Semester- I)
Subject : Computer Science Paper Code: UCSCO112
Title of Paper: DBMS-I Paper: II
Credit: 2 No. of lectures: 36

Learning objective: Students successfully completing this course will be able to:

- Understand design and implementation of a database system.
- Study the physical, logical database designs and database modeling.
- Understanding and development for essential DBMS concepts.
- Understand creations, manipulation and querying of data in databases.

Learning Outcomes:

- Master the basics of database concepts and database management system
- Model an application's data requirements using conceptual modeling tools like ER model, relational model.
- Write SQL commands to create tables, insert, update, delete and querying data.

Units	Title & Content	No. Of lecture
Unit I	**1. Introduction to File organization & DBMS** 1.1 Introduction 1.2 Types of file organization 1.3 File system Vs DBMS 1.4 Data models 1.5 Levels of abstraction 1.6 Data independence 1.7 Structure of DBMS 1.8 Users of DBMS 1.9 Advantages of DBMS	**04**
Unit II	**2.Conceptual Design (E-R model)** 2.1 Overview of DB design 2.2 ER data model (entities, attributes, entity sets, relations, relationship sets) 2.3 Additional constraints (Key constraints, Mapping constraints) 2.4 Conceptual design using ER modelling 2.4 Case studies	**10**
Unit III	**3. Relational data model** 3.1 Structure of Relational Databases (concepts of a table, a row, a relation, a Tuple, and a key in a relational database) 3.2 Conversion of ER to Relational model 3.3. Integrity constraints (primary key, referential integrity, unique constraint, Null constraint, Check constraint)	**04**
Unit IV	**4. Relational algebra** 4.1 Preliminaries 4.2 Relational algebra (selection, projection set operations, renaming, joins, division) 4.3 Problems.	**04**
Unit V	**5. Introduction to SQL** 5.1 Introduction 5.2 Basic structure 5.3 Set operations 5.4 Aggregate functions 5.5 Null values 5.6 PL/PgSqL: Data types, Language structure	**08**

Unit VI	**6. Operations with SQL** 6.1 Nested Subqueries 6.2 Modifications to Database 6.3 DDL and DML commands with examples 6.4 SQL mechanisms for joining relations (inner joins, outer joins and their types) 6.5 Examples on SQL (case studies)	**06**

References

1. Shamkant B. Navathe, Ramez Elmasri, Database Systems, ISBN:9780132144988,PEARSON HIGHER EDUCATION
2. Richard Stones, Neil Matthew, Beginning Databases with PostgreSQL: From Novice to Professional, ISBN:9781590594780, Apress
3. Korry Douglas,PostgreSQL, ISBN:9780672327568, Sams
4. JohnWorsley, Joshua Drake , Practical PostgreSQL(BCD),ISBN:9788173663925 Shroff/O'reilly
5. Joshua D. Drake, John C Worsley , Practical Postgresql , **(O'Reilly publications)**
6. Bipin C Desai , "An introduction to Database systems" , Galgotia Publications
7. Henrey Korth, Sudarshan, Silberschatz "Database System Concepts" (4th Ed), McGraw Hill,.

Paper III: UCSCO113 Lab Course I : Basic Programming Using C

Course Objectives:

Students successfully completing this course will be able:

1. Design and implement a 'C' programs for simple problems
2. Understand appropriate use of data types and array structures
3. Understand use of appropriate control structures

Course Outcomes:

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

1. Formulate pseudocodes and flowchart for computational problems.
2. Write, debug and execute simple programs in ‘C’.

Guidelines:

Lab Book: The lab book is to be used as a hands-on resource, reference and record of assignment submission and completion by the student. The lab book contains the set of assignments which the student must complete as a part of this course.

Submission: Problem Solving Assignments: The problem solving assignments are to be submitted by the student in the form of a journal containing individual assignment sheets. Each assignment includes the Assignment Title, Problem statement, Date of submission, Assessment date, Assessment grade and instructors sign.

Assessment:

Continuous assessment of laboratory work is to be done based on overall performance and lab assignments performance of student.

Operating Environment:

For ‘C’ Programming:

- Turbo C++ 4.0 Windows 7 Windows 8 64Bit Version

PAPER CODE : UCSCO113 PAPER – III : Lab Course I : Basic Programming Using C (Credits – 02 No. of Practical's – 10)	
	Title of Experiment/ Practical
1	Assignment to demonstrate use of data types, simple operators & expressions.
2	Assignment to demonstrate decision making statements (if and if-else, nested structures)
3	Assignment to demonstrate decision making statements (switch - case)
4	Assignment to demonstrate use of simple loops
5	Assignment to demonstrate use of nested loops
6	Assignment to demonstrate menu driven programs.
7	Assignment to demonstrate writing C programs in modular way (use of user defined functions)
8	Assignment to demonstrate recursive functions.
9	Assignment to demonstrate use of arrays (1-d arrays) and functions
10	Assignment to demonstrate use of arrays (2-d arrays) and functions

Class: F.Y. B. Sc. (Comp. Sci.) Practical Lab-I Course (Semester- I)

Subject: Computer Science — Paper Code: UCSCO114

Title of Paper: Lab Course on DBMS-I — Paper : II

Credit: 2 — No. of Practical's: 10

Learning objective: Students successfully completing this course will be able to:

- Understand design and implementation of a database system.
- Study the physical, logical database designs and database modeling.
- Understanding and development for essential DBMS concepts.
- Understand creations, manipulation and querying of data in databases.

Learning Outcomes:

- Master the basics of database concepts and database management system
- Write, debug, and execute SQL Queries using PostgreSQL.
- Model an application's data requirements using conceptual modeling tools like ER model, relational model.
- Write SQL commands to create tables, insert, update, delete and querying data.

Sr. No.	Title of Experiment/ Practical
1	Create simple tables, with only the primary key Constraint
2	Create more than one table with integrity constraint
3	Create more than one table, with referential integrity constraint.
4	Drop a table from database, Alter the table.
5	Insert/Update/Delete statements.
6	Query for the tables using simple form of Select Statement
7	Query solving for table operations (Aggregate function)
8	Nested Query solving for table operations (Union, Intersect, Except)
9	Nested Query solving for table operations (Set membership, Cardinality, Comparison)
10	Small Case Studies.

Paper I: UCSCO121 Advanced Programming Using C

Prerequisites:

1. Problem Solving tools like algorithms, flowcharts and pseudocodes.
2. Basic knowledge of 'C' language.

Course Objectives:

Students successfully completing this course will be able:

1. To study advanced concepts of programming using 'C' language.
2. To understand complex data types like structure and union.
3. To work with files.
4. To understand and develop basics of Graphics Programming

Course Outcomes:

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

1. Develop programs using control structures, pointers, strings, structures and files in 'C'.
2. Design and develop solutions to real world problems using C.

PAPER CODE : UCSCO121 **PAPER – I : Advanced Programming Using C** **(Credits – 02 No. of Lectures – 36)**		
	Chapter and Sub Topics	**No. of Lectures**
Unit – I	**Pointers** 1.1 Pointer declaration, initialization 1.2 Dereferencing pointers 1.3 Pointer arithmetic 1.4 Pointer to pointer 1.5 Arrays and pointers 1.6 Functions and pointers – passing pointers to functions, function returning pointers 1.7 Dynamic memory allocation	8
Unit – II	**Strings** 2.1 Declaration and initialization, format specifiers 2.2 Standard library functions 2.3 Strings and pointers	6

	2.4 Array of strings 2.5 Command Line Arguments	
Unit – III	**Structures and Unions** 3.1 Creating structures 3.2 Accessing structure members (dot Operator) 3.3 Structure initialization 3.4 Typedef 3.5 Array of structures 3.6 Passing structures to functions 3.7 Nested structures 3.8 Pointers and structures 3.9 Self referential structure 3.10 Unions 3.11 Difference between structures and unions	10
Unit – IV	**File Handling** 4.1 Streams 4.2 Types of Files 4.3 Operations on files 4.4 Random access to files	6
Unit – V	**C Preprocessor** 4.1 Format of Preprocessor directive 4.2 File Inclusion directive 4.3 Macro substitution, nested macro, argumented macro **4.4 Macros VS Functions**	2
Unit – VI	**Graphics programming using C** 6.1 Graphics driver and mode 6.2 Drawing simple graphical objects–line, circle, rectangle etc. 6.3 Outputting text, curves & Polygons	4

References:

1. Yashavant Kanetkar : Let Us C 7th Edition, PBP Publications
2. E Balaguruswamy : Programming in ANSI C 7th Edition, Tata Mc-Graw Hill Publishing Co. Ltd.-New Delhi
3. Brian W. Kernighan and Dennis M. Ritchie : The C Programming Language 2nd Edition, Prentice Hall Publication
4. The Complete Reference to C, Herbert Schildt
5. Problem Solving with C, Harrow
6. Programming in C ,A Practical Approach, Ajay Mittal , Pearson

Class: F.Y. B. Sc. (Computer Science) (Semester- II)

Subject: Computer Science Paper Code: UCSCO122

Title of Paper: DBMS-II Paper: II

Credit: 2 No. of lectures: 36

Prerequisites: Knowledge of DBMS

Learning Objectives:-Students successfully completing this course will be able to:

- Understand fundamental concepts of RDBMS (PL/PgSQL)
- Understand data security and its importance.
- Understand client server architecture.

Learning Outcomes:

- Develop the database design by normalization.
- Knowing functional dependencies and design of the relational database.
- Design concept of Transaction and Query processing.

Unit	Title & Contents	No. of lectures
Unit I	**1. Relational Database Design** 1.1 Preliminaries 1.2 Normalization (1NF,2NF,3NF,BCNF,4NF, 5 NF) 1.3 Controlling the program flow, conditional statements, loops 1.4 Handling errors and exceptions, Cursors 1.5 Views, Stored Functions, Stored Procedures, Triggers, **Index**	12
Unit II	**2 Transaction Concepts and concurrency control** 2.1 Transaction, properties of transaction, state of the transaction. 2.2 Executing transactions concurrently associated problem in concurrent execution. 2.3 Schedules, types of schedules, Serializability, precedence graph for Serializability. 2.4 Ensuring Serializability by locks, different lock modes, 2PL and its variations. 2.5Basic timestamp method for concurrency, Thomas Write Rule. 2.6 Locks with multiple granularity, dynamic database concurrency (Phantom Problem). 2.7 Timestamps versus locking. 2.8 Deadlock handling methods 2.8.1 Detection and Recovery (Wait for graph). 2.8.2 Prevention algorithms (Wound-wait, Wait-die)	10
Unit III	**3 Database Integrity and Security Concepts** 3.1 Domain constraints 3.2 Referential Integrity 3.3 Introduction to database security concepts 3.4 Methods for database security 3.4.1Discretionary access control method 3.4.2Mandatory access control and role base access control for multilevel security.	6

	3.5 Use of views in security enforcement. 3.6 Overview of encryption technique for security. 3.7 Statistical database security.	
Unit IV	**4 Crash Recovery** 4.1 Failure classification 4.2 Recovery concepts 4.3 Log base recovery techniques (Deferred and Immediate update) 4.4 Checkpoints 4.5 Recovery with concurrent transactions (Rollback, checkpoints,commit) 4.6 Database backup and recovery from catastrophic failure. 4.7 **DCL Command implementation with example (Grant & Revoke Command)**	4
Unit V	5. **Client-Server Technology** 5.1 Describe client-server computing. 5.2 Evolution of Client - Server information systems. 5.3 Client – Server Architecture benefits. 5.4 Client Server Architecture - Components, Principles, Client Components, Communication middleware components, Database middleware components, Client Server Databases	4

References:-

1. Elmasri and Navathe,Fundamentals of Database Systems (4th Ed)
2. Henrey Korth, Sudarshan, Silberschatz,Database System Concepts (4th Ed)
3. Practical PostgreSQL O"REILLY
4. Richard Stones , Neil Matthew,Beginning Databases with PostgreSQL, From Novice to Professional, 2nd Edition, Apress

Paper III: UCSCO123 Lab Course I : Advanced Programming Using C

Course Objectives:

Students successfully completing this course will be able:

1. Design and implement a real world computational problems using Advanced 'C' programming concepts.
2. Understand appropriate use of data types like structures, Union
3. Understand use of Files

Course Outcomes:

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

1. Write, debug and execute programs using advanced concepts in ‘C’.

Guidelines:

Lab Book: The lab book is to be used as a hands-on resource, reference and record of assignment submission and completion by the student. The lab book contains the set of assignments which the student must complete as a part of this course.

Submission: Problem Solving Assignments: The problem solving assignments are to be submitted by the student in the form of a journal containing individual assignment sheets. Each assignment includes the Assignment Title, Problem statement, Date of submission, Assessment date, Assessment grade and instructors sign.

Assessment:

Continuous assessment of laboratory work is to be done based on overall performance and lab assignments performance of student.

Operating Environment:

For ‘C’ Programming:

- Turbo C++ 4.0 Windows 7 Windows 8 64Bit Version

PAPER CODE : UCSCO123 PAPER – III : Lab Course I : Advanced Programming Using C (Credits – 02 No. of Practical's – 10)	
	Title of Experiment/ Practical
1	Assignment to demonstrate use of pointers.
2	Assignment to demonstrate concept of strings (string & pointers)
3	Assignment to demonstrate array of strings.
4	Assignment to demonstrate use of bitwise operators.
5	Assignment to demonstrate structures and unions.
6	Assignment to demonstrate structures (using array and functions).
7	Assignment to demonstrate command line arguments and preprocessor directives.
8	Assignment to demonstrate file handling (text files & binary files)
9	Assignment to demonstrate graphics programming.
10	C Programming – Case study

Class: F.Y. B. Sc. (Comp. Sci.) Practical Lab-II Course (Semester- II)

Subject: Computer Science — Paper Code: UCSCO124

Title of Paper: Lab Course on DBMS-II — Paper : II

Credit: 2 — No. of Practical's: 10

Learning objective: Students successfully completing this course will be able to:

- Understand design and implementation of a database system.
- Study the physical, logical database designs and database modeling.
- Understanding and development for essential RDBMS concepts.
- Understand creations, manipulation and querying of data in databases.

Learning Outcomes:

- Outline the fundamental concepts of relational Database Management System.
- Perform advanced Relational database Management Operations.
- Validate the queries by implementing error and exception handling techniques.
- Write queries, functions, triggers, cursor, and views using SQL and PL/SQL.

Sr. No.	Title of Experiment/ Practical
1	Simple Queries
2	Nested Queries, using aggregate functions
3	Queries using Views
4	Stored Function
5	Cursors
6	Exception Handling
7	Triggers
8	Case Study