Shiromani Gurudwara Parbandhak Committee's

Guru Nanak Khalsa College of Arts, Science and Commerce (Autonomous)

Matunga, Mumbai – 400 019, Maharashtra

Program: Bachelor of Science

Syllabus

Course: FYBSc

Semester I and II

(Name of Subject: Computer Science)

(As per NEP guidelines-DSC model)

With effect from Academic Year 2025 - 2026)

Guru Nanak Khalsa College of Arts, Science and Commerce (Autonomous)
Program Structure
Semester- I

Course Name	Teaching Hours		Credits Assigned			Marks		
	Theory	Practical	Theory	Practical	Total	Theory	Practical	Total
Major Paper (Digital Systems and Architecture) GNKUSCSMJ1101	30	60	2	2	4	50	50	100
Minor Paper (Fundamentals of DBMS) GNKUSCSMI1101	30	60	2	2	4	50	50	100
Open Elective I Personal Financial Planning I GNKUCBAFOE1101	30	--	2	-	2	50	--	50
Open Elective II Overview of Banking II GNKUCBBIOE2101	30	--	2	-	2	50	--	50
Vocational Skill Course (VSC) (Linux Operating System) GNKUSCSVSC101	--	60	--	2	2	--	50	50
Skill Enhancement Course (SEC) (Python Programming I) GNKUSCSSEC101	--	60	--	2	2	--	50	50
Ability Enhancement Skill (AEC) (Communication Skills in English I) GNKUSCSAEC101	30	--	2	--	2	--	50	50
Value Education Course (VEC) (Ethics and Culture I) GNKUSCSVEC101	30	--	2	--	2	--	50	50
Co-curricular (CC) GNKUSCSCC101	--	--	--	--	2	--	--	50
	180	240	12	8	22	200	300	550

Guru Nanak Khalsa College of Arts, Science and Commerce (Autonomous)
Program Structure
Semester- II

Course Name	Teaching Hours		Credits Assigned			Marks		
	Theory	Practical	Theory	Practical	Total	Theory	Practical	Total
Major Paper (Design & Analysis of Algorithms) GNKUSCSMJ1102	30	60	2	2	4	50	50	100
Minor Paper (Introduction to OOPs using C++) GNKUSCSMI1102	30	60	2	2	4	50	50	100
Open Elective I (Personal Financial Planning II) GNKUCBAFOE1102	30	--	2	-	2	50	--	50
Open Elective II (Overview of Banking II)GNKUCBBIOE2102	30	--	2	-	2	50	--	50
Vocational Skill Course (VSC) (E-Commerce & Digital Marketing) GNKUSCSVSC102	--	60	--	2	2	--	50	50
Skill Enhancement Course (SEC) (Statistics with R Programming) GNKUSCSSEC102	--	60	--	2	2	--	50	50
Ability Enhancement Skill (AEC) (Communication Skills in English II) GNKUSCSAEC102	30	--	2	--	2	--	50	50
Value Education Course (VEC) (Ethics and Culture II) GNKUSCSVEC102	30	--	2	--	2	--	50	50
Indian Knowledge System (IKS) (Title of the Paper) GNKUSCSIKS102	--	--	--	--	2	50	--	50
	180	180	12	6	22	250	300	550

Guru Nanak Khalsa College of Arts, Science and Commerce (Autonomous)
Department of Arts

PROGRAMME OUTCOMES (PO)

BACHELOR OF SCIENCE (BSc)

Undergraduate Science Program Outcomes:

PO1	**Foundational Understanding:** Develop a foundational understanding of core scientific principles and theories across various disciplines of science.
PO2	**Analytical Skills:** Develop analytical and problem-solving skills to critically analyse scientific problems and apply scientific methodologies.
PO3	**Global Perspective:** Gain a global perspective by understanding diverse scientific issues and incorporating ethical considerations in scientific practices.
PO4	**Research Awareness:** Gain awareness of research methodologies and techniques, preparing for future research endeavours.
PO5	**Holistic Development:** Experience holistic development by embracing values of humanism, empathy, and social responsibility in scientific pursuits.
PO6	**Communication Skills:** Enhance communication skills to effectively convey scientific concepts to diverse audiences.
PO7	**Continuous Learning:** Develop a commitment to lifelong learning and staying updated with advancements in science.
PO8	**Ethical Practices:** Understand and adhere to ethical standards in scientific research and practice.

Guru Nanak Khalsa College of Arts, Science and Commerce (Autonomous)
Department of Computer Science

Programme: BSc CS

Programme Specific Outcomes (PSOs) for BSc in Computer Science

Sr. No.	A student completing BSc in CS will be able to:
PSO 1	Gain comprehensive understanding across various domains of information technology to facilitate advanced studies and research.
PSO 2	Cultivate analytical and problem-solving proficiencies essential for tackling real-world challenges within the field.
PSO 3	Master the art of communicating technical concepts and designs effectively to diverse audiences.
PSO 4	Foster collaborative teamwork skills, emphasizing constant communication and cooperation to develop innovative projects.
PSO 5	Apply contemporary tools and techniques proficiently to analyze system concepts and available data, enabling informed decision-making.
PSO 6	Demonstrate proficiency in reading and utilizing programming language documentation to develop, modify, and optimize systems for practical applications.
PSO 7	Develop adept technical writing skills tailored to effectively communicate information technology-related concepts.
PSO 8	Acquire expertise in database programming, encompassing design, implementation, and management of databases
PSO 9	Attain proficiency in developing predictive and clustering models, leveraging statistical and machine learning techniques for data analysis.Acquire skills in data visualization to present complex information in a clear and insightful manner.
PSO 10	Establish foundational knowledge in digital logic to understand fundamental principles underlying computer systems and architectures.

Guru Nanak Khalsa College of Arts, Science and Commerce (Autonomous)
Department of Computer Science

Course: F.Y.B.Sc. CS
Semester-I: Major Paper I
Course Title: Digital Systems and Architecture
Course Code: GNKUSCSMJ1101
Credits: 2
No of lectures (Hours): 30
Marks: 50

Course Objectives:

Sr. No.	Course objectives
The course aims at:	
1	To compare and analyze computer organization and architecture, focusing on system components, their functions, and interconnection structures, including buses, I/O modules, interrupt mechanisms, and DMA.
2	Understand and apply the fundamentals of digital logic design, including logic gates, simplification techniques such as truth tables and Karnaugh Maps, and the construction of combinational and sequential circuits like adders, multiplexers, flip-flops, and counters.
3	This course explores the organization and hierarchy of memory systems, covering internal (RAM, SRAM, DRAM), cache, virtual memory, and external (magnetic, optical, flash) memory technologies, along with RAID configurations for performance and reliability.
4	To examine the design and functionality of instruction set architectures, memory addressing, instruction sequencing, branching, and addressing modes, and to distinguish between RISC and CISC architectures concerning performance and complexity.
5	To introduce the concepts of advanced computer architectures, including classifications of parallel systems through Flynn's taxonomy, and explore the working of array processors, clustered systems, and Non-Uniform Memory Access (NUMA) models.

Course Outcomes (COs):

Sr. No.	On completing the course, the student will be able to:	POs addresse	PSOs addresse	Cognitive Levels

		d	d	addressed
CO 1	Describe and differentiate the concepts of computer organization and architecture, including major system components and interconnection structures such as buses, I/O modules, interrupts, and DMA.	PO1, PO7	PSO 1, PSO2	R, U
CO 2	Apply digital logic concepts to design and simplify logical expressions using truth tables and Karnaugh Maps, and construct basic combinational and sequential circuits such as adders, multiplexers, flip-flops, and counters. Understanding Number system and its conversions.	PO2, PO3, PO4	PSO 3, PSO 4, PSO 5	Ap, An
CO 3	Demonstrate understanding of memory system organization, including internal and external memory types, cache, virtual memory, and RAID levels, and evaluate their roles in overall system performance.	PO3, PO4, PO5	PSO 5, PSO 6,	E
CO 4	Analyze instruction set architecture, memory addressing, instruction sequencing, branching, and addressing modes, and compare RISC and CISC architectures based on design and performance.	PO5, PO6, PO7, PO8	PSO 6, PSO 7, PSO 10	An, C
CO 5	Explain the fundamentals of advanced computer architecture and parallel processing systems, including Flynn's taxonomy, array processors, clustered systems, and NUMA models.	PO1, PO7, PO8	PSO 8, PSO 9, PSO 10	U, Ap

Unit		**Title**	**No. of lectures**	**CO Mapping**
Unit 1			**15**	
	1.1	Computer System: Comparison of Computer Organization & Architecture, Computer Components and Functions, Interconnection Structures. Bus Interconnections, Input /		**CO 1**

		Output: I/O Module, Interrupt Driven I/O, Direct Memory Access.		
	1.2	Fundamentals of Digital Logic: Logic Gates, Simplification of Logical expression using truth table, Karnaugh Maps. Number system and its conversions.		**CO 2**
	1.3	Combinational Circuits: Adders, Mux, De-Mux, Sequential Circuits: Flip Flops (SR, JK & D), Counters: synchronous and asynchronous Counter		**CO 3**
Unit 2			**15**	
	2.1	Memory System Organization: Internal Memory: RAM, SRAM and DRAM. Cache Memory, Virtual Memory, External Memory.		**CO 4**
	2.2	Instruction Set Architecture, Memory location and Addresses, Memory Operations. Instruction and Instruction Sequencing, Branching, Addressing Modes, Basic Input / Output, Interrupts. Introduction to RISC and CISC Architecture.		**CO 5**
	2.3	Fundamentals of Advanced Computer Architecture: Parallel Architecture: Classification of Parallel Systems, Flynn's Taxonomy, Array Processors, Clusters, and NUMA Computers.		**CO 5**

References:

1. M. Mano, Computer System Architecture 3rd edition, Pearson
2. Carl Hamacher et al., Computer Organization and Embedded Systems, 6 ed., McGraw-Hill
   2012
3. R P Jain, Modern Digital Electronics, Tata McGraw Hill Education Pvt. Ltd. , 4th Edition, 2010
4. William Stallings (2010), Computer Organization and Architecture- designing for performance,8th edition, Prentice Hall, New Jersy.
5. Anrew S. Tanenbaum (2006), Structured Computer Organization, 5th edition, PearsonEducation Inc,
6. John P. Hayes (1998), Computer Architecture and Organization, 3rd edition, Tata McGrawHill

Examination:

Total Marks: 50 Marks

- **Internal Examination (20 Marks):** 8 Marks exam (MCQ and short answer question) with 20% completed syllabus. Duration of exam: 15 minutes. 7 Marks for either of

Quiz/Assignments/Presentation/Viva and 5 Marks for Overall Performance (Class Participation, Attendance)

- **End Semester theory examination (30 Marks):** Weightage of each unit will be proportional to the number of lecture hours as mentioned in the syllabus. Duration of exam: 1 hours
- **Combined passing of 40% with minimum 20% in Internal Component.**

Course: FY B.Sc. CS Practical

Semester-I: Major

Course Title: Digital Systems and Architecture

Course Code: GNKUSCSMJ1P101

Credits: 02

No of Practical (Hours): 60

Marks: 50

Course Objectives:

Sr. No.	Course objectives
The course aims at:	
1	To introduce the fundamental logic gates (NOT, AND, OR, NAND, NOR, EX-OR, EX-NOR) and enable students to study and verify their truth tables through circuit implementation.
2	To develop the ability to simplify Boolean expressions and implement them using logic gates for efficient circuit design.
3	To design and verify basic combinational circuits, including half/full adders, subtractors, comparators, and logic expressions using multiplexers and demultiplexers.
4	To understand and implement sequential logic circuits, including flip-flops (SR, JK, D), counters (synchronous/asynchronous), and shift registers.
5	Apply digital design principles to build and test complex circuits, such as ripple counters and other modular designs, using a combination of combinational and sequential components.

Course Outcomes (COs):

Sr. No.	On completing the course, the student will be able to:	POs addressed	PSOs addressed	Cognitive Levels addressed
CO 1	Demonstrate the ability to construct and verify the truth tables of logic gates and implement simplified Boolean expressions using logic circuits.	PO1, PO2, PO6	PSO 1, PSO 2	R, U
CO 2	Design and test combinational circuits such as half/full adders, half/full subtractors, and 4-bit magnitude comparators.	PO1, PO2, PO4, PO7	PSO 2, PSO 3, PSO 4	Ap, An
CO 3	Implement and analyze the behavior of multiplexers and demultiplexers for realizing logical expressions.	PO3, PO4, PO7, PO8	PSO 5, PSO 6	An, E

CO 4	Design, construct, and verify the operations of sequential circuits, including SR, JK, and D flip-flops.	PO4, PO5, PO6	PSO 5, PSO 7, PSO 8	An, C
CO 5	Build and evaluate the performance of counters and shift registers, including a 3-bit ripple counter using JK flip-flops and a 4-bit shift register.	PO5, PO7, PO8	PSO 9, PSO 10	C, R

List of Experiments:

1. Study and verify the truth table of various logic gates:
    I. Basic Logic gates
        A. NOT Gate
        B. AND Gate
        C. OR Gate
    II. Universal gates
        A. NAND Gate
        B. NOR Gate
    III. Exclusive gates
        A. XOR Gate
        B. XNOR Gate
2. Design and verify working of NAND gates as OR gate.
3. Implement NAND gates as AND gate, validate it's working.
4. Design and validate working of NAND gates as XOR gate.
5. Construct an XNOR gate from NAND gate.
6. Design and verify working of NOR gates as OR gate.
7. Implement NOR gates as AND gate, validate it's working.
8. Design and validate working of NOR gates as XOR gate.
9. Construct XNOR gate from NOR gate.
10. Simplify given Boolean expression using logic gates and realize it using the truth table.
11. Design and verify a half adder
12. Implement a circuit to demonstrate working of half subtractor.
13. Construct a circuit to illustrate the operation of a full adder / full subtractor.
14. Design a 4 bit magnitude comparator using combinational circuits.
15. Implement a logic circuit to simulate the behavior of SR flip-flop.
16. Construct a circuit to demonstrate a counter and verify its operations.
17. Build and test a circuit to show the operation of a 4 bit shift register
18. Design and implement working of multiplexers.
19. Set up a circuit to explain the working mechanism of demultiplexers.
20. Construct a circuit to illustrate the operation of a 3-bit binary ripple counter using JK flip flops.

References:

1. R P Jain, Modern Digital Electronics, Tata McGraw Hill Education Pvt. Ltd. , 4th Edition, 2010

2. William Stallings (2010), Computer Organization and Architecture- designing for performance,8th edition, Prentice Hall, New Jersy.

Total Marks: 50 Marks

- **Experiment Marks: 40 Marks**
- **Journal & Viva Marks: 5 + 5 Marks**

Guru Nanak Khalsa College of Arts, Science and Commerce (Autonomous)
Department of Computer Science

Course: F.Y.B.Sc. CS
Semester-I: Minor
Course Title: Fundamentals of DBMS
Course Code: GNKUSCSMI1101
Credits: 2
No of lectures (Hours): 30
Marks: 50

Course Objectives:

Sr. No.	Course objectives
The course aims at:	
1	To understand the fundamental concepts of databases and DBMS architecture, including data models, levels of abstraction, and data independence.
2	To develop the skills for designing conceptual database models using ER modeling and converting ER diagrams to relational schemas.
3	To learn and apply Data Definition Language (DDL) and Data Manipulation Language (DML) statements for creating and managing database objects and performing basic CRUD operations.
4	To implement relational database principles using functions, joins, subqueries, and normalization techniques for efficient data organization and retrieval.
5	To apply advanced database concepts including views, user access control, database security, and transaction control commands.

Course Outcomes (COs):

Sr. No.	On completing the course, the student will be able to:	POs addressed	PSOs addressed	Cognitive Levels addressed
CO 1	Describe the basic concepts of database systems, DBMS architecture, data models, and advantages of using DBMS.	PO1, PO2	PSO1	R,, Un
CO 2	Design an Entity-Relationship (ER) model and transform it into a relational schema with appropriate keys and constraints.	PO2, PO3, PO5	PSO1, PSO2	Un, Ap
CO 3	Apply DDL and DML commands to create, alter, update, and manage tables and perform CRUD operations effectively	PO1, PO2, PO5	PSO1, PSO3	Ap

CO 4	Write SQL queries involving aggregate functions, conditional selections, joins, and subqueries to retrieve and manipulate data from multiple tables.	PO1, PO4, PO5	PSO1, PSO3	Ap,An
CO 5	Normalize relational schemas up to BCNF and demonstrate lossless join decomposition for optimized database structure.	PO2, PO3	PSO1	Ap
CO 6	Implement advanced SQL features including functions, views, and indexing to optimize and organize data access.	PO1, PO5, PO6	PSO1, PSO3	Ap,An
CO 7	Demonstrate user management, backup/restore, transaction control, and security mechanisms using DCL commands.	PO6, PO7, PO8	PSO2	Ap

Unit		**Title**	**No. of lectures**	**CO Mapping**
Unit 1			**15**	CO1, CO2, CO3
	1.1	**Introduction to DBMS**: Database, DBMS – Definition, Overview of DBMS, Advantages of DBMS, Levels of abstraction, Data independence, DBMS Architecture **Data models**: Client/Server Architecture, Object Based Logical Model, Record Based Logical Model (relational, hierarchical, network Entity Relationship		CO1
	1.2	**Model and ER to Table**: Entities, attributes, entity sets, relations, relationship sets, Additional constraints (key constraints, participation constraints, weak entities, aggregation / generalization, Conceptual Design using ER (entities VS attributes, Entity Vs relationship, binary Vs ternary, constraints beyond ER) Entity to Table, Relationship to tables with and without key constraints.		CO2
	1.3	**DDL Statements**: Creating Databases, Using Databases, datatypes, Creating Tables (with integrity constraints – primary key, default, check, not null), Altering Tables, Renaming Tables, Dropping Tables, Truncating Tables **DML statements**: Viewing the structure of a table insert, update, delete, Select all columns, specific columns, unique records, conditional select, in clause,		CO3

		between clause, limit, aggregate functions (count, min, max, avg, sum), group by clause, having clause		
Unit 2			**15**	CO3,CO4, CO5
	2.1	**Relational data model**: Domains, attributes, Tuples and Relations, Relational Model Notation, Characteristics of Relations, Relational Constraints - primary key, referential integrity, unique constraint, Null constraint, Check constraint **Functions**: String Functions (concat, instr, left, right, mid, length, lcase/lower, ucase/upper, replace, strcmp, trim, ltrim, rtrim), Math Functions (abs, ceil, floor, mod, pow, sqrt, round, truncate) Date Functions(adddate, datediff, day, month, year, hour, min, sec, now, reverse)		CO4
	2.2	**Joining Tables and Subqueries**: inner join, outer join (left outer, right outer, full outer) subqueries with IN, EXISTS, subqueries restrictions, Nested subqueries, ANY/ALL clause, correlated subqueries **Normal forms:** Functional dependencies, first, second, third, and BCNF normal forms based on primary keys, lossless join decomposition.		CO4,CO3,CO5
	2.3	**Database Protection:** Security Issues, Threats to Databases, Security Mechanisms, Role of DBA, Discretionary Access Control, Backing Up and Restoring databases **Views**: Creating, altering dropping, renaming and manipulating views **DCL Statements**: Creating/dropping users, privileges introduction, granting/revoking privileges, viewing privileges), Transaction control commands – Commit, Rollback		CO5

References:

1. Fundamentals of Database System, ElmasriRamez, NavatheShamkant, Pearson Education, Seventh edition, 2017
2. Database Management Systems, Raghu Ramakrishnan and Johannes Gehrke, 3rd Edition,2014
3. Murach's MySQL, Joel Murach, 3rd Edition, 3rd Edition, 2019
4. Database System Concepts, Abraham Silberschatz, HenryF.Korth, S.Sudarshan, McGraw Hill,2017
5. MySQL: The Complete Reference, VikramVaswani , McGraw Hill, 2017

6. Learn SQL with MySQL: Retrieve and Manipulate Data Using SQL Commands with Ease, Ashwin Pajankar, BPB Publications, 2020

Examination:

Total Marks: 50 Marks

- **Internal Examination (20 Marks):** 8 Marks exam (MCQ and short answer question) with 20% completed syllabus. Duration of exam: 15 minutes. 7 Marks for either of Quiz/Assignments/Presentation/Viva and 5 Marks for Overall Performance (Class Participation, Attendance)
- **End Semester theory examination (30 Marks):** Weightage of each unit will be proportional to the number of lecture hours as mentioned in the syllabus. Duration of exam: 1 hours
- **Combined passing of 40% with minimum 20% in Internal Component.**

Course: FY B.Sc. CS Practical

Semester-I: Minor

Course Title: Fundamentals of DBMS

Course Code: GNKUSCSMI1P101

Credits: 02

No of Practical (Hours): 60

Marks: 50

Course Objectives:

Sr. No.	Course objectives
The course aims at:	
1	To understand and apply conceptual database design using Entity-Relationship (ER) modeling techniques. Students will identify entities, relationships, attributes, and constraints, and transform them into a relational schema.
2	To develop the ability to create, manage, and manipulate relational databases using SQL.Students will perform operations like creating databases and tables, modifying structures, and executing CRUD operations.
3	To write and execute SQL queries involving aggregate functions, built-in functions, joins, and subqueries. Students will use SQL to retrieve data using complex query logic with functions and nested queries.
4	To normalize database schemas and ensure data consistency and minimal redundancy. Students will convert ER models to tables and apply normalization techniques up to the Third Normal Form (3NF).
5	To understand and implement advanced SQL concepts such as views, indexing, and transaction control.Students will create and manage views, practice data control commands, and use commit/rollback effectively.
6	To comprehend the importance of database security and user management using Data Control Language (DCL).Students will apply security controls through privileges and permissions in multi-user environments.

Course Outcomes (COs):

Sr. No.	On completing the course, the student will be able to:	POs addressed	PSOs addressed	Cognitive Levels addressed
CO 1	Understand and apply ER modeling to design conceptual schemas and convert them into relational models.	PO1, PO2, PO3	PSO1, PSO2	U, Ap

CO 2	Create and manage relational databases using SQL commands for table creation, modification, and basic operations.	PO1, PO5	PSO1, PSO3	Ap
CO 3	Develop and execute SQL queries involving functions, joins, and subqueries to retrieve data from multiple tables.	PO1, PO2, PO4	PSO1, PSO3	Ap, An
CO 4	Normalize relational database schemas up to 3NF to reduce redundancy and improve data integrity.	PO2, PO3, PO5	PSO1	Ap, An
CO 5	Implement advanced SQL concepts like views, indexing, and transaction control for efficient and secure data handling.	PO1, PO4, PO5, PO6	PSO1, PSO3	Ap,E
CO 6	Apply data control commands for database security, user access control, and demonstrate transaction management techniques like COMMIT and ROLLBACK.	PO6, PO7, PO8	PSO2	Ap,U

List of Experiments:

1. ER Diagram Design :Design an ER diagram identifying entities, attributes, keys, and relationships for a system like Library, Hospital, or University.

2. Generalization and Specialization:Apply generalization and specialization in an ER diagram (e.g., Employee → {Manager, Clerk}).

3. View All Databases:Write SQL to display a list of all existing databases in your RDBMS (e.g., `SHOW DATABASES;` in MySQL).

4. Create a Database:Create a new database named `project_db` using SQL.

5. View All Tables in a Database:Display all tables from a selected database using SQL commands.

6. Create Tables with Constraints :Create a table `student` using appropriate datatypes and constraints (Primary Key, Not Null, Default, Check).

7. Create Tables Without Constraints:Create a simple table `department` without using any constraints.

8. Perform CRUD Operations:Insert, update, delete, and retrieve data from a table named `employee`.

9. Alter Table Structure:Alter the `student` table to add a new column `email` and modify data type of the `name` column.

10. Drop, Truncate, and Rename Tables:Write SQL to rename a table, truncate its data, and drop it completely.

11. Backup and Restore Operations:Demonstrate how to take a backup of a table and restore it (manually or with export/import tools).

12. Execute Basic Queries:Fetch all records from a table and apply `WHERE` conditions, ordering, and limiting.

13. Use Aggregate Functions:Apply aggregate functions like `COUNT`, `SUM`, `AVG`, `MIN`, and `MAX` in SQL queries.

14. Use Date Functions in Queries:Use date functions such as `NOW()`, `DATEDIFF()`, `YEAR()`, `MONTH()`, and `DAY()` in queries.

15. Use String and Math Functions:Use string functions (`UPPER()`, `LENGTH()`, `REPLACE()`) and math functions (`ROUND()`, `FLOOR()`, `MOD()`).

16. Execute Join Queries:Perform `INNER JOIN`, `LEFT OUTER JOIN`, `RIGHT OUTER JOIN`, and `FULL OUTER JOIN` on two or more tables.

17. Subqueries Using IN and EXISTS:Write subqueries using `IN` and `EXISTS` clauses to filter data based on related tables.

18. ER Model to Relational Conversion & Normalization:Convert a conceptual ER diagram to relational tables and normalize them up to 3rd Normal Form (3NF).

19. Create and Use Views:Create views with and without `WITH CHECK OPTION`, and use them in `SELECT` queries. Drop a view using SQL.

20. DCL and Transaction Control:Use `GRANT` and `REVOKE` to manage user permissions. Demonstrate `COMMIT` and `ROLLBACK` with insert/update actions.

References:

1. Fundamentals of Database System, ElmasriRamez, NavatheShamkant, Pearson Education, Seventh edition, 2017
2. Database System Concepts, Abraham Silberschatz, HenryF.Korth, S.Sudarshan, McGraw Hill,2017

Total Marks: 50 Marks

- **Experiment Marks: 40 Marks**
- **Journal & Viva Marks: 5 + 5 Marks**

Guru Nanak Khalsa College of Arts, Science and Commerce (Autonomous)
Department of Computer Science

Course: FY B.Sc. CS Practical

Semester-I: VSC

Course Title: Linux

Course Code: GNKUSCSVSC101

Credits: 02

No of Practical (Hours): 60

Marks: 50

Course Objectives:

Sr. No.	Course objectives
The course aims at:	
1	To provide hands-on experience in installing and configuring the Ubuntu Linux operating system.
2	To enable students to effectively use Linux-based software management tools and explore open-source applications.
3	To familiarize students with essential Linux commands for file handling, text processing, and system administration.
4	To develop skills in using terminal-based editors and scripting tools such as `vi`, `sed`, and `awk`.
5	To introduce shell scripting concepts for task automation, including conditionals, loops, and input/output operations.
6	To create a basic development environment and write simple programs using C and Python on the Ubuntu platform.

Course Outcomes (COs):

Sr. No.	On completing the course, the student will be able to:	POs addressed	PSOs addressed	Cognitive Levels addressed
CO 1	Install, configure, and manage Ubuntu Linux OS and essential software packages.	PO1, PO2	PSO1	R, Ap

CO 2	Use and administer the Linux terminal to execute system commands, handle files, and manage processes.	PO1, PO2, PO5	PSO1	U, Ap
CO 3	Demonstrate proficiency in file editing, text processing, and working with scripting tools (`vi`, `sed`, `awk`).	PO2, PO3, PO5	PSO2	Ap, An
CO 4	Write and execute shell scripts to automate system tasks using loops, conditionals, and input/output redirection.	PO1, PO2, PO3	PSO2	Ap,An
CO 5	Install compilers and write basic programs in C and Python within the Linux environment.	PO1, PO3	PSO3	Un, Ap

List of Experiments:

1. Installation of Ubuntu Linux Operating System via USB/DVD
2. Software Installation Using Ubuntu Software Center and Synaptic Package Manager
3. Exploration and Installation of Commonly Used Software Packages
4. System Administration and User Management in Ubuntu
5. Using the Terminal: Basic Commands and Navigation
6. Running Windows Programs on Ubuntu Using Wine/PlayOnLinux
7. File and Directory Management Using File System Commands
8. File Handling Operations Using Linux Commands
9. Working with File Permissions and Compression Utilities
10. Using General Purpose Linux Utility Commands
11. Text Processing with Filters and Input/Output Redirection
12. Editing Files Using vi/vim Editor
13. Text Manipulation Using sed and awk Commands
14. Process Management and Job Scheduling in Linux

15. Shell Scripting I: Variables, Input, Exit Status, and Conditionals
16. Shell Scripting II: Loops and Arithmetic Operations
17. Shell Scripting III: Input/Output Redirection and Custom Scripts
18. Shell Scripting IV: Creating Interactive and Menu-driven Scripts
19. Installation of C, C++, Java, and Python Compilers in Ubuntu
20. Basic Programming in C and Python on the Ubuntu Platform

References:

Textbooks:

1. "Linux Command line and Shell Scripting Bible", Richard Blum, Wiley India.
2. "Unix: Concepts and Applications", Sumitabha Das, 4th Edition, McGraw Hill.
3. "Official Ubuntu Book", Matthew Helmke& Elizabeth K. Joseph with Jose Antonio Rey and Philips Ballew, 8th Ed.

Additional References:

1. "Linux Administration: A Beginner's Guide", Fifth Edition, Wale Soyinka, Tata McGraw-Hill, 2008.
2. "Linux: Complete Reference", Richard Petersen, 6th Edition, Tata McGraw-Hill
3. "Beginning Linux Programming", Neil Mathew, 4th Edition, Wiley Publishing, 2008.
4. "Linux Command Line and Shell Scripting Bible" *by Richard Blum and Christine Bresnahan*
   – Comprehensive guide covering Linux commands, file systems, scripting, and process management.
5. "UNIX and Shell Programming" *by B. A. Forouzan and Richard F. Gilberg*
   – Ideal for foundational knowledge of Unix/Linux commands and shell scripting basics.
6. "Beginning Ubuntu Linux"*by Keir Thomas and Andy Channelle*
   – Step-by-step guide for Ubuntu installation, user administration, and application setup.
7. "Learning the bash Shell"*by Cameron Newham (O'Reilly)*
   – Focused book on bash scripting, covering both basics and advanced features.

Total Marks: 50 Marks

- **Experiment Marks: 40 Marks**
- **Journal & Viva Marks: 5 + 5 Marks**

Guru Nanak Khalsa College of Arts, Science and Commerce (Autonomous)
Department of Computer Science

Course: FY B.Sc. CS Practical
Semester-I: SEC
Course Title: Python Programming I
Course Code: GNKUSCSSEC101
Credits: 02
No of Practical (Hours): 60
Marks: 50

Course Objectives:

Sr. No.	Course objectives
The course aims at:	
1	To learn how to design and program Python applications.
2	To understand components of the Python Program.
3	To define the structure and components of a Python program.
4	To learn how to write loops and decision statements in Python.
5	To learn about inbuilt input/output operations and compound data types in Python.

Course Outcomes (COs):

Sr. No.	On completing the course, the student will be able to:	POs addressed	PSOs addressed	Cognitive Levels addressed
CO 1	Ability to define the structure and components of a Python program.	PO1 PO2	PSO1,PSO3	Ap, An
CO 2	Ability to implement basic Input / Output operations in Python	PO1 PO2	PSO1,PSO2	U,Ap
CO 3	Ability to implement operators in Python	PO1 PO2	PSO1,PSO2	U,Ap, An

CO 4	Ability to learn how to write loops and decision statements in Python.	PO1 PO2 PO7	PSO1,PSO2	Ap, An
CO 5	Ability to learn how to use Array in Python.	PO1 PO2 PO7	PSO1,PSO2	Ap, An
CO 6	Ability to learn how to write functions and pass arguments in Python.	PO1 PO2	PSO1,PSO2,PSO5	Ap, An
CO 7	Ability to learn how to write strings and string functions in Python.	PO1 PO2	PSO1,PSO2	Ap, An,C
CO 8	Ability to create and use Compound data types in Python	PO1 PO2 PO7	PSO1,PSO2,PSO5	Ap, An,C
CO 9	Ability to create and use dictionaries in Python	PO1 PO2 PO7	PSO1,PSO2	Ap, An,C

List of Experiments:

1. Introduction to python IDLE.
2. Setting up a Python development environment.
3. Executing basic python statements.
4. Write a program to design and develop basic python programs.
5. Write a program to demonstrate various data types in python.
6. Write a program to implement basic Input / Output operations in Python.
7. Write a Python program to demonstrate the various types of operators in python.
8. Write a program to design and develop a python program to implement various conditional statements using suitable examples.
9. Write a program to design and develop a python program to implement various looping statements using suitable examples.
10. Write a python program to create and manipulate arrays in Python.
11. Write a program to demonstrate use of slicing and indexing for accessing elements from the array.
12. Write a program in Python to define and call functions for suitable problems.
13. Write a Python program to demonstrate different types of function arguments.
14. Write a Python program to implement and use lambda function in python
15. Write a Python program to implement String and String functions in python.
16. Write a program to implement a list in Python for suitable problems. Demonstrate

various operations on it.

17. Write a program to implement tuples in Python for suitable problems. Demonstrate various operations on it.
18. Write a program to implement a dictionary in Python for suitable problems. Demonstrate various operations on it.
19. Write a program to implement built-in modules in Python for suitable problems.
20. Write a program to implement user defined modules in Python for suitable problems.

References:

Textbooks:

1. Practical Programming: An Introduction to Computer Science Using Python 3, Paul Gries , Jennifer Campbell, Jason Montojo, Pragmatic Bookshelf, 2nd Edition, 2014
2. Programming through Python, M. T Savaliya, R. K. Maurya & G M Magar, Sybgen Learning India, 2020

Additional References:

1. Python: The Complete Reference, Martin C. Brown, McGraw Hill, 2018
2. Beginning Python: From Novice to Professional, Magnus Lie Hetland, Apress, 2017
3. Programming in Python 3, Mark Summerfield, Pearson Education, 2nd Ed, 2018
4. Python Programming: Using Problem Solving Approach, ReemaThareja, Oxford Univeristy Press, 2017
5. Let Us Python, Yashwant. B. Kanetkar, BPB Publication, 2019

Total Marks: 50 Marks

- **Experiment Marks: 40 Marks**
- **Journal & Viva Marks: 5 + 5 Marks**

Guru Nanak Khalsa College of Arts, Science and Commerce (Autonomous)
Department of Computer Science

Course: F.Y.B.Sc. CS
Semester-I: VEC
Course Title: Ethics and Culture I
Course Code: GNKUSCSVEC101
Credits: 2
No of lectures (Hours): 30
Marks: 50

Course Objectives:

Sr. No.	Course objectives
The course aims at:	
1	The course aims to help students explore ethical and cultural dimensions of their lives.
2	The course provides a forum for students to pause, revisit their assumptions and beliefs, and become mindful of their thoughts, emotions and actions.
3	It gives the students an opportunity to express themselves and inquire into their decision-making processes. This will enable them to cultivate ethical values and participate in the creation of a society based on acceptance, compassion, and justice

Course Outcomes (COs):

Sr. No.	On completing the course, the student will be able to:	POs addressed	PSOs addressed	Cognitive Levels addressed
CO 1	Explore perspectives on ethics in thoughts, words and actions	PO1 ,PO2	PSO1, PSO3	Ap, An
CO 2	Evolve ethical decision making practises	PO1, PO2	PSO1,PSO2	U,Ap
CO 3	Evolve ethical decision making practises	PO1, PO2	PSO1,PSO 2	U,Ap, An

CO 4	Understand the need for an ethical society and culture			

Unit		Title	No. of lectures	CO Mapping
Unit 1			**15**	
	1.1	**Introduction - The Basis of Ethics:** Meaning of ethics, Nature of ethical values, Importance of ethics, Ethics in daily life. **Getting to Know Each Other:** Self-awareness, Understanding others, Values and beliefs, Mutual respect. **What to Expect from the Course?** Learning outcomes, Ethical awareness, Skill development. **Recognition of Our Common Humanity:** Common human values, Human dignity, Equality and respect, social responsibility. **Empathy, Compassion and Justice:** Importance of empathy, Compassion as a value, Justice and fairness, Social harmony		**CO1**
Unit 2			**15**	
	2.1	**The Role of Intelligence, Reason and Emotions Discernment:** **What Is the Right Thing To Do?:** Moral choice, Responsibility **The Art of Conflict Resolution:** Nature of conflict, Types of conflict, Resolution skills **Destructive and Constructive Emotions:** Difference between destructive and constructive emotions, Emotional impact **The Need for Emotional Balance:** meaning of Emotional balance, Self-control, Emotional intelligence, Well-being		**CO2, CO3**

References:

Textbooks:

- Aristotle. Nichomachean Ethics. London: Penguin Classics, 2004
- Swami Vivekananda. The Complete Works of Swami Vivekananda. Advaita Ashrama, 2016 --https://www.ramakrishnavivekananda.info/vivekananda/complete_works.html
- Panch Parmeshwar in English translation as The Holy Panchayat by Munshi Premchand

- The Silas Marner by George Eliot
- We are Seven by Wordsworth
- The Chimney Sweeper by William Blake

Examination:

- **Internal Examination (20 Marks):** Continuous Internal assessment (CIA) of 20 Marks; Any two among Quiz, Assignment, Presentation, Viva of 7 and 8 Marks and Class Participation of 5 marks.
- **Semester End Theory Examination (30 Marks):** Weightage of each unit will be proportional to the number of lecture hours as mentioned in the syllabus. Duration of the exam is 1 hour.
- **Combined passing of 40% with minimum 20% in Internal Component.**

Guru Nanak Khalsa College of Arts, Science and Commerce (Autonomous)
Department of Computer Science

Course: F.Y.B.Sc. CS
Semester-I: CC
Course Title: NSS/Sports/Cultural/Music
Course Code: GNKUSCSCC101
Credits: 2
No of lectures (Hours): 30
Marks: 50

Guru Nanak Khalsa College of Arts, Science and Commerce (Autonomous)
Department of Computer Science

Course: F.Y.B.Sc. CS
Semester-II: Major Paper I
Course Title: Design & Analysis of Algorithms
Course Code: GNKUCCSMJ1102
Credits: 2
No of lectures (Hours): 30
Marks: 50

Course Objectives:

Sr. No.	Course objectives
The course aims at:	
1	To make students understand the basic principles of algorithm design
2	To give idea to students about the theoretical background of the basic data structures
3	To familiarize the students with fundamental problem-solving strategies like searching, sorting, selection, recursion and help them to evaluate efficiencies of various algorithms.
4	To teach students the important algorithm design paradigms and how they can be used to solve various real world problems.

Course Outcomes (COs):

Sr. No.	On completing the course, the student will be able to:	POs addressed	PSOs addressed	Cognitive Levels addressed

CO 1	Students should be able to understand and evaluate efficiency of the programs that they write based on performance of the algorithms used.	PO2 PO4 PO7	PSO2 PSO5 PSO6	Understanding, Analyzing, Evaluating
CO 2	Students should be able to appreciate the use of various data structures as per need	PO2 PO6 PO8	PSO2 PSO5 PSO8	Understanding, Applying, Analyzing
CO 3	To select, decide and apply appropriate design principle by understanding the requirements of any real life problems	PO1 PO3 PO5 PO6	PSO1 PSO3 PSO5 PSO9 PSO10	Understanding, Applying, Analyzing, Evaluating

Unit		**Title**	**No. of lectures**	**CO Mapping**
Unit 1			**15**	
	1.1	**Introduction to Data Structures -** What is data structure, types, Introduction to Array(1-d & 2-d), Stack and List data structures, operations on these data structures, advantages disadvantages and applications of these data structures like solving linear equations, Polynomial Representation, Infix-to-Postfix conversion		CO1 CO2 CO3
	1.2	**Recursion -** What is recursion, Recursion vs Iteration, recursion applications like Factorial of a number, Fibonacci series & their comparative analysis with respect to iterative version, Tower of Hanoi, Problem		CO1 CO2 CO3

	1.3	**Introduction to algorithms** - What is algorithm, analysis of algorithm, Types of complexity, Running time analysis, How to Compare Algorithms, Rate of Growth, Types of Analysis, Asymptotic Notation, Big-O Notation, Omega-Ω Notation, Theta-Θ Notation, Asymptotic Analysis, Performance characteristics of algorithms, Estimating running time / number of steps of executions on paper, Idea of Computability		CO1 CO3
Unit 2			**15**	
	2.1	**Algorithm Design Techniques -** Introduction to various types of classifications/design criteria and design techniques		CO1 CO3
	2.2	**Greedy Technique -** Concept, Advantages & Disadvantages, Applications, Implementation using problems like - file merging problem **Divide-n-Conquer -** Concept, Advantages & Disadvantages, Applications, Implementation using problems like - merge sort, Strassen's Matrix Multiplication		CO1 CO2 CO3
	2.3	**Dynamic Programming** - Concept, Advantages & Disadvantages, Applications, Implementation using problems like - Fibonacci series, Factorial of a number, Longest Common subsequence. **Backtracking Programming -** Concept, Advantages & Disadvantages, Applications, Implementation using problems like N-Queen Problem		CO1 CO2 CO3

References:

1. "Data Structure and Algorithm Using Python", Rance D. Necaise, Wiley India Edition, 2016.

2. “Data Structures and Algorithms Made Easy”, NarasimhaKarumanchi, CareerMonk Publications, 2016.
3. “Introduction to Algorithms”, Thomas H. Cormen, 3rd Edition, PHI.
4. “Introduction to the Design and Analysis of Algorithms”, Anany Levitin, Pearson, 3rd Edition, 2011.
5. “Design and Analysis of Algorithms”, S. Sridhar, Oxford University Press, 2014.

Examination:

Total Marks: 50 Marks

- **Internal Examination (20 Marks):** 8 Marks exam (MCQ and short answer question) with 20% completed syllabus. Duration of exam: 15 minutes. 7 Marks for either of Quiz/Assignments/Presentation/Viva and 5 Marks for Overall Performance (Class Participation, Attendance)
- **End Semester theory examination (30 Marks):** Weightage of each unit will be proportional to the number of lecture hours as mentioned in the syllabus. Duration of exam: 1 hours
- **Combined passing of 40% with minimum 20% in Internal Component.**

Course: FY B.Sc. CS Practical
Semester-II: Major
Course Title: Design & Analysis of Algorithms
Course Code: GNKUSCSMI1P102
Credits: 02
No of Practical (Hours): 60
Marks: 50

Course Objectives:

Sr. No.	**Course objectives**
The course aims at:	
1	To make students understand the basic principles of algorithm design
2	To give idea to students about the theoretical background of the basic data structures
3	To familiarize the students with fundamental problem-solving strategies like searching, sorting, selection, recursion and help them to evaluate efficiencies of various algorithms.
4	To teach students the important algorithm design paradigms and how they can be used to solve various real world problems.

Course Outcomes (COs):

Sr. No.	**On completing the course, the student will be able to:**	**POs addressed**	**PSOs addressed**	**Cognitive Levels addressed**
CO 1	Students should be able to understand and evaluate efficiency of the programs that they write based on performance of the algorithms used.	PO2 PO4 PO7	PSO2 PSO5 PSO6	Understanding, Analyzing, Evaluating

CO 2	Students should be able to appreciate the use of various data structures as per need	PO2 PO6 PO8	PSO2 PSO5 PSO8	Understanding, Applying, Analyzing
CO 3	To select, decide and apply appropriate design principle by understanding the requirements of any real life problems	PO1 PO3 PO5 PO6	PSO1 PSO3 PSO5 PSO9 PSO10	Understanding, Applying, Analyzing, Evaluating

List of Experiments:

Sr. No.	**Title**
	Programs on 1-d arrays like - i. sum of elements of array ii. searching an element in array
	Programs on 1-d arrays like - i. finding minimum and maximum element in array ii. count the number of even and odd numbers in array
	Programs on 2-d arrays like i. row-sum ii. column-sum iii. sum of diagonal elements

Programs on 2-d arrays like

i. sum of diagonal elements

ii. addition of two matrices

iii. multiplication of two matrices

5 Program to create a list-based stack and perform various stack operations.

6 Program to perform linear search and binary search on list of elements.

7 Program to perform binary search on list of elements on list of elements.

8 Programs to sort elements of list by using algorithms bubble sort.

9 Programs to sort elements of list by using algorithms selection sort.

10 Programs to sort elements of list by using algorithms insertion sort.

11 Programs to sort elements of list by using algorithms merge sort.

12 Programs on recursion like factorial & fibonacci.

13 Programs on recursion like sum of n numbers & tower of hanoi.

14 Program to implement N-Queen problem using Backtracking.

15 Program to implement fibonacci series using Dynamic Programming.

16 Write a python program to generate all permutations of a given string or list using Dynamic Programming.

17 Program to implement Floyd-Warshall algorithm using Dynamic Programming.

18 Program to implement coin change problems using Greedy Approach.

19 Program to implement Dijkstra's Algorithm using Greedy Approach.

20	Given fuel stations along a route, find the minimum number of stops to reach the destination using the Greedy algorithm.

References:

6. "Data Structure and Algorithm Using Python", Rance D. Necaise, Wiley India Edition, 2016.
7. "Data Structures and Algorithms Made Easy", NarasimhaKarumanchi, CareerMonk Publications, 2016.
8. "Introduction to Algorithms", Thomas H. Cormen, 3rd Edition, PHI.
9. "Introduction to the Design and Analysis of Algorithms", Anany Levitin, Pearson, 3rd Edition, 2011.
10. "Design and Analysis of Algorithms", S. Sridhar, Oxford University Press, 2014.

Total Marks: 50 Marks

- **ü Experiment Marks: 40 Marks**
- **ü Journal & Viva Marks: 5 + 5 Marks**

Guru Nanak Khalsa College of Arts, Science and Commerce (Autonomous)
Department of Computer Science

Course: F.Y.B.Sc. CS
Semester-II: Minor
Course Title: Introduction to OOPs using C++
Course Code: GNKUSCSMI1102
Credits: 2
No of lectures (Hours): 30
Marks: 50

Course Objectives:

Sr. No.	Course objectives
The course aims at:	
1	To introduce the fundamental concepts of object-oriented programming and its advantages over procedural programming.
2	To familiarize students with the syntax, structure, and features of the C++ programming language.
3	To develop problem-solving skills using control structures, functions, arrays, and strings in C++.
4	To impart knowledge of advanced object-oriented concepts like constructors, inheritance, polymorphism, and file handling.
5	To enable students to model and implement real-life problems using UML diagrams and OOP principles effectively.

Course Outcomes (COs):

Sr. No.	On completing the course, the student will be able to:	POs addressed	PSOs addressed	Cognitive Levels addressed
CO 1	Understand the principles of object-oriented programming and distinguish it from procedural programming.	PO1, PO2	PSO1	R, U
CO 2	Apply C++ syntax and programming constructs to write and execute basic programs using input/output, variables, and control structures.	PO1, PO3	PSO1	Ap
CO 3	Demonstrate the use of arrays, strings, and functions in problem-solving.	PO1, PO2, PO3	PSO2	Ap,An

CO 4	Design classes and implement encapsulation, abstraction, and constructors to structure C++ programs effectively.	PO2, PO3, PO4	PSO1, PSO2	Ap,An
CO 5	Implement inheritance, polymorphism, and operator overloading to develop modular and reusable code.	PO3, PO4, PO5	PSO2	An, E
CO 6	Use pointers and file handling techniques to manage memory and persistent data in applications.	PO1, PO3	PSO2	Ap,An
CO 7	Model real-world applications using object-oriented principles and UML class diagrams.	PO4, PO5, PO6	PSO2, PSO3	Ap,C

Unit		**Title**	**No. of lectures**	**CO Mapping**
Unit 1			**15**	CO1,CO2
	1.1	**Introduction to Programming Concepts**: Object oriented programming paradigm, basic concepts of object oriented programming, benefits of object oriented programming, object oriented languages, applications of object oriented programming.Tokens-keywords, identifiers, constants-integer, real, character and string constants, backslash constants, features of C++ and its basic structure, simple C++ program without class, compiling and running C++ program. **Data Types, Data Input Output and Operators**: Basic data types,variables, rules for naming variables, programming constants, the type cast operator, implicit and explicit type casting, cout and cin statements,operators, precedence of operators.		CO1,C02
	1.2	**Decision Making**, Loops, Arrays and Strings: Conditional statements-if,if...else, switch loops- while, do...while, for, types of arrays and string and string manipulations **Unified Modeling Language (UML)**: Introduction to UML & class diagrams. **Classes, Abstraction & Encapsulation**: Classes and objects, Dot Operator, data members, member functions, passing		CO1,CO2

		data to functions, scope and visibility of variables in function.		
	1.3	**Constructors and Destructors**: Default constructor, parameterized constructor, copy constructor, private constructor, destructors. **Working with objects**: Accessor - mutator methods, static data and static function, access specifiers, array of objects.		CO4,CO5
Unit 2			**15**	CO3,CO4,CO5,C06
	2.1	**Polymorphism** - Binding-static binding & overloading, constructor overloading function overloading, operator overloading, overloading unary and binary operators. **Modelling Relationships in Class Diagrams**: Association, AggregationComposition and examples covering these principles **Inheritance:** Defining base class and its derived class, access specifiers, types of inheritance-single, multiple, hierarchical, multilevel, hybrid inheritance, friend function and friend class, constructors in derived classes.		CO3, CO4
	2.2	**Modelling Relationships**: Generalization-Specialization and examples covering these principles **Run time Polymorphism** - Dynamic Binding, Function overriding, virtual function, pure virtual function, virtual base class, abstract class. **Pointers**: Introduction to pointers, * and & operators, assigning addresses to pointer variables, accessing values using pointers, pointers to objects & this pointer, pointers to derived classes		CO3, CO5
	2.3	**File Handling**: File Stream classes, opening and closing file-file opening modes, text file handling, binary file handling. **Applying OOP to solve real life applications**: To cover case studies like library management, order management etc. to design classes covering all relationships		CO5, CO6

References:

1. Object Oriented Programming with C++, Balagurusamy E., 8th Edition, McGraw Hill Education India.

2. UML & C++: A Practical Guide to Object Oriented Development, Lee/Tepfenhart, Pearson Education, 2nd Edition2015
3. Mastering C++ by Venugopal, Publisher: McGraw-Hill Education, 2017
4. Let Us C++ by KanetkarYashwant, Publisher: BPB Publications, 2020
5. Object Oriented Analysis and Design by Timothy Budd TMH, 2001

Examination:

Total Marks: 50 Marks

- **Internal Examination (20 Marks):** 8 Marks exam (MCQ and short answer question) with 20% completed syllabus. Duration of exam: 15 minutes. 7 Marks for either of Quiz/Assignments/Presentation/Viva and 5 Marks for Overall Performance (Class Participation, Attendance)
- **End Semester theory examination (30 Marks):** Weightage of each unit will be proportional to the number of lecture hours as mentioned in the syllabus. Duration of exam: 1 hours
- **Combined passing of 40% with minimum 20% in Internal Component.**

Course: FY B.Sc. CS Practical

Semester-II: Minor

Course Title: Introduction to OOPs using C++

Course Code: GNKUSCSMI1P102

Credits: 02

No of Practical (Hours): 60

Marks: 50

Course Objectives:

Sr. No.	Course objectives
The course aims at:	
1	Introducing the fundamentals of object-oriented programming and class structure.
2	Developing the ability to design and implement programs using classes and objects.
3	Exploring control structures (branching and looping) within class member functions.
4	Understanding the concepts of constructors, destructors, and access specifiers.
5	Demonstrating different types of inheritance and their practical implementations.
6	Enhancing programming skills through advanced features like function overloading, friend functions, file handling, and pointer usage in classes.

Course Outcomes (COs):

Sr. No.	On completing the course, the student will be able to:	POs addressed	PSOs addressed	Cognitive Levels addressed
CO 1	Understand the basic concepts of object-oriented programming and apply them to create and use simple classes and objects.	PO1, PO2	PSO1	R, U
CO 2	Implement control structures (if, else, loops) within class methods to design logical solutions.	PO1, PO3	PSO1	Ap, An

CO 3	Demonstrate the use of arrays, scope resolution, and access specifiers within classes.	PO2, PO5	PSO1	Un, Ap
CO 4	Construct programs using constructors, destructors, and demonstrate constructor overloading.	PO2, PO3	PSO2	Ap, An
CO 5	Implement different types of inheritance and examine their effects on class design and constructor invocation.	PO1, PO3, PO4	PSO2	An,E
CO 6	Develop advanced programs using friend functions, pointers, file handling, and demonstrate function	PO1, PO3, PO5	PSO3	Ap,C

List of Experiments:

1. Create a simple class with data members and member functions. Demonstrate object creation and function invocation.
2. Write a program to accept and display student details using a class and object.
3. Develop a program that performs basic arithmetic operations using class methods and if-else branching.
4. Create a class to check whether a number is prime or not using loops within class methods.
5. Write a program using a class that calculates factorial and Fibonacci series using loops.
6. Create a class that handles one-dimensional arrays and performs operations like sum, average, and max.
7. Develop a class that uses a two-dimensional array to store and process a matrix (e.g., addition, transpose).
8. Illustrate the use of scope resolution operator by declaring and accessing variables at different scopes (global, class, local).
9. Write a program to demonstrate default constructor and its usage.
10. Implement a class with parameterized constructor and show how to initialize objects with different values.
11. Create a program that demonstrates copy constructor and explain when it is invoked.
12. Write a class that demonstrates the use of destructor and tracks when the object is destroyed.

13. Create a class that demonstrates the use of public, private, and protected access specifiers.
14. Implement single inheritance to show how a derived class can access base class members.
15. Write a program to demonstrate multilevel inheritance and constructor invocation order.
16. Develop a program that uses multiple inheritance to combine features of two base classes.
17. Demonstrate hierarchical inheritance using a base class and multiple derived classes.
18. Write a program using a friend function to access private data of two different classes.
19. Demonstrate the use of inline function and `this` pointer within a class.
20. Create a class that uses pointers for dynamic memory allocation and includes file read/write (text and binary).

References:

1. Data Structure and Algorithm Using Python, Rance D. Necaise, Wiley India Edition, 2016.
2. Object Oriented Programming with C++, Balagurusamy E., 8th Edition, McGraw Hill Education India.
3. Data Structures and Algorithms Made Easy, Narasimha Karumanchi, CareerMonk Publications, 2016.
4. Let Us C++ by KanetkarYashwant, Publisher: BPB Publications, 2020

Total Marks: 50 Marks

- **Experiment Marks: 40 Marks**
- **Journal & Viva Marks: 5 + 5 Marks**

Course: FY B.Sc.CS Practical

Semester-II:VSC

Course Title: E-Commerce and Digital Marketing

Course Code: GNKUSCSVSC102

Credits: 02

No of Practical (Hours): 60

Marks: 50

Course Objectives:

Sr. No.	Course objectives
The course aims at:	
1	To make learners aware about basic probability axioms and rules and its application.
2	To understand the concept of conditional probability and Independence of events.
3	To make learners familiar with discrete and continuous random variables as well as standard discrete and continuous distributions.
4	To learn computational skills to implement various statistical inferential approaches.

Course Outcomes (COs):

Sr. No.	On completing the course, the student will be able to:	POs addressed	PSOs addressed	Cognitive Levels addressed
CO 1	Understand the core concepts of E-Commerce.	PO1,PO2	PSO1, PSO2	U,An
CO 2	Understand the various online techniques	PO1,PO4	PSO1, PSO2, PSO7	U,An
CO 3	Understand the core concepts of digital marketing and the role of digital marketing in business.	PO1,PO4,PO7	PSO3, PSO4,PSO7	U,An,Ap
CO 4	Apply digital marketing strategies to increase sales and growth of business	PO1,PO4,PO6, PO7	PSO4,PSO5,PSO7	Ap,E, C
CO 5	Apply digital marketing through different channels and platforms	PO1,PO2,PO6, PO7	PSO4,PSO5,PSO7	Ap, E, C

CO 6	Understand the significance of Web Analytics and Google Analytics and apply the same.	PO1,PO2,PO7	PSO1, PSO2, PSO7	U,An,Ap

List of Experiments:

1. Digital Marketing Basics
    a. Create and compare a traditional vs digital marketing strategy for a local business.
    b. List the differences between SEO, SEM, and SMM with examples.
2. Digital Marketing tools
3. Digital Marketing Implementation in Business Scenario
4. Social Media Marketing (SMM)
    a. List and categorize 10 social media platforms (e.g., Facebook = Networking, YouTube = Video sharing).
    b. Create audience personas for a specific product or brand.
    c. Analyze 3 competitor social media pages and summarize their engagement strategy.
5. Create the Digital Marketing Webpage
6. Create a website for a business.
7. Conducting the Search Engine Optimization and Search Engine Marketing
8. Using Google Analytics to analyze website performance
9. Creating Promotional banner through Canva
10. Creating brochures for business or events through Canva
11. Facebook Promotion using banners
12. Facebook Promotion using story
13. Creating YouTube Channel for Marketing
    a. Create a branded YouTube channel
    b. Add a profile picture and banner (channel art)
    c. Write a compelling channel description with keywords
    d. Add links to your website/social media
14. Creating and adding Video on youtube channel
15. Twitter Marketing
16. Instagram Marketing
17. Email Marketing
18. Content Marketing
    a. Write a blog post targeting a specific keyword.
    b. Design a content calendar for a month.
    c. Create SEO-friendly titles and meta descriptions.

19. Create and send a welcome email sequence.
20. Use Instagram Insights to track:
    a. Post reach & impressions
    b. Engagement rate
    c. Profile visits & website clicks
    d. Follower growth

References:

Textbooks:

1. "E-Commerce Strategy, Technologies and Applications", Whitley, David, Tata McGraw Hill, 2017
2. Digital Marketing, Seema Gupta, McGraw Hill Education, 2nd Edition

Additional References:

1. E-Commerce by S. Pankaj, A.P.H. Publication, New Delhi
2. Fundamentals of Digital Marketing, Punit Singh Bhatia, Pearson, 2nd Edition
3. "Understanding Digital Marketing: MarketingStrategies for Engaging the Digital Generation", Damian Ryan, Calvin Jone. Kogan Page, 4th Edition

Total Marks: 50 Marks

- **Experiment Marks: 40 Marks**
- **Journal & Viva Marks: 5 + 5 Marks**

Course: FY B.Sc.CS Practical

Semester-II:SEC

Course Title: : Statistics with R Programming

Course Code: GNKUSCSSEC102

Credits: 02

No of Practical (Hours): 60

Marks: 50

Course Objectives:

Sr. No.	Course objectives
The course aims at:	
1	To understand and apply built-in statistical functions in R for data analysis.
2	To perform linear and multiple regression analysis for modeling and prediction.
3	To explore probability distributions (normal and binomial) using R functions.
4	To analyze and visualize time series data for identifying patterns and trends.
5	To construct, analyze, and interpret contingency tables and cross tabulations.
6	To create and interpret various graphical representations for single and multivariable data.

Course Outcomes (COs):

Sr. No.	On completing the course, the student will be able to:	POs addressed	PSOs addressed	Cognitive Levels addressed
CO 1	Apply built-in statistical functions in R to summarize and describe datasets effectively.	PO1, PO2	PSO1, PSO2	U,Ap
CO 2	Conduct linear and multiple regression analysis using R and interpret the results for decision-making.	PO2, PO5	PSO2	An,Ap
CO 3	Utilize R functions to compute and interpret normal and binomial distributions in real-world contexts.	PO1, PO3	PSO1, PSO2	U,Ap

CO 4	Create, analyze, and visualize time series data to identify trends and seasonal variations.	PO2, PO4	PSO2	An,E
CO 5	Construct contingency and cross-tabulation tables and derive meaningful insights from categorical data.	PO1, PO2, PO6	PSO1	An,E
CO 6	Generate various types of plots (single, two-variable, multi-variable) and store/export graphics for reporting.	PO3, PO5, PO10	PSO1, PSO2	Ap,C

1. Install R and RStudio on your system.
   After installation, write and execute a simple R script that prints "Hello, R World!" using RStudio.
2. Customize RStudio according to your preferences.
   Change the font size, editor theme, and default working directory. Capture a screenshot of your customized setup.
   Use R GUI to perform a basic statistical summary.
3. Open the built-in `mtcars` dataset and use the GUI options to generate a summary of the data.Create and save an R Script in RStudio that contains a program to take user input for name and age, and print a greeting message with both.

4. Looping Constructs in R

Implement `for`, `while`, and `repeat` loops to process numeric sequences or perform iterative tasks (e.g., factorial, sum of series).

5. Working with Date and Time in R

Write a program to fetch the current system date/time.

Format and manipulate date and time objects using functions like `Sys.Date()`, `as.Date()`, `strptime()`.

5. Creating and Manipulating Vectors and Matrices

Create numeric, character, and logical vectors.

Perform operations like indexing, slicing, and vector arithmetic.

Construct and operate on matrices.

6. Working with Arrays, Lists, and Data Frames

Create a 3D array and access elements.

Build a list containing vectors and data frames.

Create a data frame from vectors and apply functions like `summary()`, `str()`, `nrow()`, `ncol()`.

7. Writing and Using Functions in R

Define user-defined functions with parameters and return values.

Demonstrate built-in functions and use of anonymous functions with `apply()` or `sapply()`.

8. Implementing Character Strings in R

Create and manipulate character strings.

Use functions like `nchar()`, `toupper()`, `tolower()`, `substr()`, `paste()`, and `strsplit()`.

9. String Operations and Pattern Matching

Perform string searching and pattern matching using `grep()`, `grepl()`, `sub()`, and `gsub()`.

Demonstrate regular expressions in filtering and replacing data.

10.String Operations and Pattern Matching.

Perform string searching and pattern matching using `grep()`, `grepl()`, `sub()`, and `gsub()`.

Demonstrate regular expressions in filtering and replacing data.

11. Descriptive Statistics

Question: Given a numeric vector `scores <- c(78, 85, 92, 70, 88, 95, 79)`, calculate the mean, median, and standard deviation of the scores using built-in functions in R.

12. Linear Regression

Question:

Use the built-in `mtcars` dataset to perform linear regression to predict `mpg` based on `hp` (horsepower). Display the regression summary and plot the regression line.

13. Multiple Regression

Question:

Perform a multiple regression using the `mtcars` dataset to predict `mpg` using `hp`, `wt`, and `cyl`. Interpret the coefficients and determine the most significant predictor.

14. Normal Distribution

Question:

For a normal distribution with a mean of 50 and standard deviation of 10:

- Find the probability that a value is less than 60 using `pnorm()`.
- Generate 100 random values using `rnorm()` and plot a histogram.

15. Binomial Distribution

Question:
If a coin is flipped 10 times, calculate the probability of getting exactly 6 heads using `dbinom()`. Then simulate 50 such experiments using `rbinom()` and plot the result.

16. Time Series Analysis

Question:
Create a time series object in R for monthly sales data:
`sales <- c(220, 240, 260, 300, 280, 310, 330, 350, 370, 360, 390, 400)`
Plot the time series and use basic time series functions like `ts()` and `plot()`.

17. Contingency Tables

Question:
Use the `Titanic` dataset available in R. Create a contingency table showing the survival count by gender and class using `table()` and `xtabs()`.

18. Cross Tabulation

Question:
Generate a cross-tabulation using a dataset with categorical variables (like `mtcars$cyl` and `mtcars$gear`). Use `table()` and visualize it with `mosaicplot()`.

19. Graphical Models: Two-Variable Plot

Question:
Plot the relationship between `wt` and `mpg` from the `mtcars` dataset. Use `plot()`, add a regression line using `abline()`, and customize plot labels.

20. Special Plots and Storing Graphics

Question:
Create a boxplot of `mpg` grouped by `cyl` in `mtcars`, then store the plot as a PNG file using `png()` and `dev.off()` functions.

References:

1. "The Art of R Programming" by Norman Matloff

A comprehensive guide for beginners and intermediate learners covering data structures, functions, and object-oriented features in R.

2. "R for Data Science" by Hadley Wickham and Garrett Grolemund

Focuses on practical data science workflows in R using tidyverse packages—ideal for real-world applications and visualization.

3. "Hands-On Programming with R" by Garrett Grolemund

A hands-on guide for writing functions, using loops, and managing control structures like conditions in R.

4"Beginning R: The Statistical Programming Language" by Mark Gardener

Good for understanding data structures, basic statistics, and GUI tools like R Commander and RStudio.

5. *The R Book by Michael J. Crawley A comprehensive reference covering data analysis and statistical techniques using R.*

6. *Introductory Statistics with R by Peter Dalgaard*

Great for understanding statistical concepts with practical R implementation.

7. *R Documentation*

Official help and reference pages for all R functions. Access via ?function_name in R or online at:

https://www.rdocumentation.org

Total Marks: 50 Marks

- **Experiment Marks: 40 Marks**
- **Journal & Viva Marks: 5 + 5 Marks**

Guru Nanak Khalsa College of Arts, Science and Commerce (Autonomous)
Department of Computer Science

Course: F.Y.B.Sc. CS
Semester-II: VEC
Course Title: Ethics and Culture II
Course Code: GNKUSCSVEC102
Credits: 2
No of lectures (Hours): 30
Marks: 50

Course Objectives:

Sr. No.	Course objectives

The course aims at:	
1	The course aims to help students explore ethical and cultural dimensions of their lives.
2	The course provides a forum for students to pause, revisit their assumptions and beliefs, and become mindful of their thoughts, emotions and actions.
3	It gives the students an opportunity to express themselves and inquire into their decision-making processes. This will enable them to cultivate ethical values and participate in the creation of a society based on acceptance, compassion, and justice

Course Outcomes (COs):

Sr. No.	On completing the course, the student will be able to:	POs addressed	PSOs addressed	Cognitive Levels Addressed
CO 1	Understand the need for an ethical society and culture	PO6, PO8	PSO1	U, An
CO 2	Introspect, become conscious of and assess one's stance in life	PO6	PSO2	A , An
CO 3	Cultivate empathy, tolerance and compassion	PO6, PO8, PO9, PO10	PSO1, PSO2	A, E
CO 4	Apply the values learnt in the course to everyday life.	PO6, PO8,	PSO1, PSO2	A , C

Unit		Title	No. of lectures	CO Mapping
Unit 1			**15**	
	1.1	**Cultivating Inner Values- Ethics in the World of Work and Play** **Training the Mind:** Mindfulness and Kindness, Practice and Impact **Meditation:** Ethical consideration and cultural impact **Discovering your Vocation and Interests**: Meaning of vocation and interests, Importance, Steps to Discover Self-discipline, Integrity, Commitment, Creativity **Work-Life Balance:** Ways to Achieve Work–Life Balance		CO 1
	Unit 2			**15**

	2.1	**Striving for a Better World I Outreach Activities:** Meaning, Types of Outreach Activities and Their Impact **Means and Ends:** Means and Ends in Ethical Decision-Making and Personal Decision-Making **Debate and Dialogue:** Difference Between Debate and Dialogue, Importance of Respectful Communication **Culture as Shared Values:** Values as a Core Component of Culture, Expression of Shared Values, Culture in a Globalized World **Creating and Sustaining Ethical Cultures:** The Role of Philosophy, Religion, Literature, Theatre, Cinema, Music, Media; Outreach Activities		CO 2, CO3

References:

Textbooks:

- Aristotle. Nichomachean Ethics. London: Penguin Classics, 2004
- Swami Vivekananda. The Complete Works of Swami Vivekananda. Advaita Ashrama, 2016 --https://www.ramakrishnavivekananda.info/vivekananda/complete_works.html
- Panch Parmeshwar in English translation as The Holy Panchayat by Munshi Premchand
- The Silas Marner by George Eliot
- We are Seven by Wordsworth
- The Chimney Sweeper by William Blake

Examination:

- **Internal Examination (20 Marks):** Continuous Internal assessment (CIA) of 20 Marks; Any two among Quiz, Assignment, Presentation, Viva of 7 and 8 Marks and Class Participation of 5 marks.
- **Semester End Theory Examination (30 Marks):** Weightage of each unit will be proportional to the number of lecture hours as mentioned in the syllabus. Duration of the exam is 1 hour.
- **Combined passing of 40% with minimum 20% in Internal Component.**