Shiromani Gurudwara Prabandhak Committee's

Guru Nanak Khalsa College of Arts, Science and Commerce (Autonomous)
Matunga, Mumbai – 400 019, Maharashtra

Program: Bachelor of Science
Syllabus

Course: SY B.Sc. Computer
Science Semester III and IV
(Major paper-I/II/Minor/OE/VSC/AEC/SEC/VEC)

(As per NEP guidelines-DSC model)
With effect from Academic Year 2024 - 2025)

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

Program Structure

Semester-III

Course Code	Course Name	Teaching Hours		Credits Assigned		
		Theory	**Practical**	**Theory**	**Practical**	**Total**
GNKUSCSMJ1103	(Major) Paper-I Operating System	45	30	3	1	4
GNKUSCSMJ2103	(Major) Paper-II Core Java	45	30	3	1	4
GNKUSCSMI103	(Minor) Paper Web Technologies	45	30	3	1	4
GNKUSCSOE103	Open elective (OE) Scientific Programming-I	30	--	2	--	2
GNKUSCSVSC103	Vocational Skill Course (VSC) PL/SQL	--	60	--	2	2
GNKUSCSAEC103	Ability Enhancement Course (AEC) Graph Theory	30	---	2	--	2
	Co-curricular (CC)	--	--	--	--	2
GNKUSCSOJT/FP/ CC/RP/CEP103	On job training (OJT)/ Field project (FP)/Research project (RP)/ Community engagement & service (CEP)	--	--	--	--	2
Total		**195**	**150**	**13**	**05**	**22**

Semester-IV

Course Code	Course Name	Teaching Hours		Credits Assigned		
		Theory	**Practical**	**Theory**	**Practical**	**Total**
GNKUSCSMJ1104	(Major) Paper-I Theory of Computation	45	30	3	1	4
GNKUSCSMJ2104	(Major) Paper-II Web Programming with Java	45	30	3	1	4
GNKUSCSMI104	(Minor) Paper Python Programming I	45	30	3	1	4
GNKUSCSOE104	Open elective (OE) Scientific Programming-II	30	--	2	--	2
GNKUSCSSEC104	Skill Enhancement Course (SEC) IoT Technology	--	60	--	2	2
GNKUSCSAEC104	Ability Enhancement Course (AEC) Software Engineering	30	--	2	--	2
	Co-curricular (CC)	--	--	--	--	2
GNKUSCSOJT/FP/ CC/RP/CEP104	On job training (OJT)/ Field project (FP)/Research project (RP)/ Community engagement & service (CEP)	--	--	--	--	2
Total		**195**	**150**	**13**	**05**	**22**

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

Department of Computer Science

SEMESTER-III

Course: SY B.Sc. Computer Science
Semester-III Paper-I
Course Title: Operating System
Course Code: GNKUSCSMJ1103
Credits: 3
No of lectures (Hours): 45
Marks: 100 (75:25)

Course Objectives:

Sr. No.	Course objectives
The course aims at:	
1	Learners must understand proper working of operating system. To provide a sound understanding of the Computer operating system, its structures, functioning and algorithms.
2	Explain various components of computer hardware and how processes are working in operating systems.
3	Discuss the structure of the operating system, its functions and algorithms.
4	Understanding the working of operating system, its structures and functioning
5	Compare various algorithms used in operating systems.

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 will comprehend operating system fundamentals, including resource management and process handling, and demonstrate proficiency in applying these concepts to analyze, troubleshoot, and optimize computing environments	PO1,PO2,PO4	PSO1,PSO2, PSO5	U , AP.
CO 2	students will understand operating system services,	PO1,PO2,PO4	PSO1,PSO2, PSO3	U , AP

	user interfaces, and system calls, enabling them to design, implement, and evaluate operating system structures for diverse computing environments, emphasizing efficiency and usability.			
CO 3	students will grasp the concepts of processes and threads, including scheduling, operations, and interprocess communication, enabling them to design and implement efficient multiprocessing and multithreading applications in diverse computing environments.	PO1,PO2,PO4	PSO1,PSO2.	U , AP
CO 4	students will understand the principles of process synchronization, including race conditions and critical-section problems, and be proficient in implementing synchronization mechanisms such as mutex locks, semaphores, and Peterson's solution to mitigate concurrent access issues and solve classic synchronization problems efficiently.	PO1,PO2,PO4	PSO1,PSO2.	U , AP
CO 5	Students will be able to analyze, compare, and implement various CPU scheduling algorithms, considering different scheduling criteria and thread scheduling techniques to optimize system performance in diverse computing environments.	PO1,PO2,PO4	PSO1,PSO2, PS05	U , AP
CO 6	students will demonstrate proficiency in analyzing, identifying, and employing methods for preventing, avoiding, detecting, and recovering from deadlocks in operating systems, ensuring system reliability and performance.	PO1,PO2,PO4.	PSO1,PSO2.	U , AP
CO 7	Students will acquire the ability to understand and apply main memory management concepts, including address translation, swapping, memory allocation techniques like contiguous allocation, segmentation, and paging, along with the structure and functioning of page tables to optimize memory utilization in operating systems.	PO1,PO2,PO4.	PSO1,PSO2, PS05	U , AP
CO 8	Students will gain proficiency in virtual memory management, including demand paging, copy-on-write, and page replacement, as well as understanding mass-storage structures like disk scheduling and file-system interfaces, enabling efficient storage allocation and access in operating systems.	PO1,PO2,PO4.	PSO1,PSO2, PS05	U , AP
CO 9	students will demonstrate competence in designing and implementing file systems, including directory structures, allocation methods, and free-space management techniques, ensuring efficient and reliable	PO1,PO2,PO4.	PSO1,PSO2.	U , AP

	storage management in operating systems.			

Unit		Title	No. of lectures	CO Mapping
Unit 1			**15**	
	1.1	Introduction and Operating-Systems Structures: Definition of Operating system, Operating System's role, Operating-System Operations, Functions of Operating System, Computing Environments		**CO 1**
	1.2	Operating-System Structures: Operating-System Services, User and Operating-System Interface, System Calls, Types of System Calls, Operating-System Structure		**CO 2**
	1.3	Processes: Process Concept, Process Scheduling, Operations on Processes, Interprocess Communication Threads: Overview, Multicore Programming, Multithreading Models		**CO 3**
Unit 2			**15**	
	2.1	Process Synchronization: General structure of a typical process, race condition, The Critical-Section Problem, Peterson's Solution, Synchronization Hardware, Mutex Locks, Semaphores, Classic Problems of Synchronization, Monitors		**CO 4**
	2.2	CPU Scheduling: Basic Concepts, Scheduling Criteria, Scheduling Algorithms (FCFS, SJF, SRTF, Priority, RR, Multilevel Queue Scheduling, Multilevel Feedback Queue Scheduling), Thread Scheduling		**CO 5**
	2.3	Deadlocks: System Model, Deadlock Characterization, Methods for Handling Deadlocks, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection, Recovery from Deadlock		**CO 6**
Unit 3			**15**	
	3.1	Main Memory: Background, Logical address space, Physical address space, MMU, Swapping, Contiguous Memory Allocation, Segmentation, Paging, Structure of the Page Table		**CO 7**

	3.2	Virtual Memory: Background, Demand Paging, Copy-on-Write, Page Replacement, Allocation of Frames, Thrashing Mass-Storage Structure: Overview, Disk Structure, Disk Scheduling, Disk Management File-System Interface: File Concept, Access Methods, Directory and Disk Structure, File-System Mounting, File Sharing		**CO 8**
	3.3	File-System Implementation: File-System Structure, File-System Implementation, Directory Implementation, Allocation Methods, Free- Space Management		**C0 9**

Textbook(s):

1. Abraham Silberschatz, Peter Galvin, Greg Gagne, Operating System Concepts, Wiley,8th Edition

Additional Reference(s):

1. Achyut S. Godbole, Atul Kahate, Operating Systems, Tata McGraw Hill

2. Naresh Chauhan, Principles of Operating Systems, Oxford Press

3. Andrew S Tanenbaum, Herbert Bos, Modern Operating Systems, 4e Fourth Edition,Pearson Education, 2016

Examination:

- **Internal Examination (25 Marks):** 20 Marks exam (MCQ and short answer question) with 20% completed syllabus. Duration of exam: 40 minutes. And 5 Marks for either Quiz/Assignments /Class Participation etc.
- **End Semester theory examination (75 Marks):** Weightage of each unit will be proportional to the number of lecture hours as mentioned in the syllabus. Duration of exam: 2hours 30mins
- **Combined passing of 40% with minimum 20% in Internal Component.**

Course: SY B.Sc. Computer Science
Practical Semester-III
Course Title: Operating System
Practical Paper-I
Course Code: GNKUSCSMJ1P103
Credits: 01
No of Practical (Hours): 30
Marks: 50

Course Objectives:

Sr. No.	Course objectives
The course aims at:	
1	Gain practical experience in implementing key operating system algorithms and mechanisms, including process scheduling, memory management, and synchronization.
2	Develop proficiency in programming concepts related to scheduling algorithms, page replacement, deadlock detection, and resource allocation.
3	Enhance problem-solving skills by applying fundamental operating system principles to real-world programming tasks.
4	Deepen understanding of operating system concepts through hands-on implementation and analysis of algorithms in diverse scenarios.

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 proficiency in implementing a variety of operating system algorithms, including scheduling, memory management, and resource allocation, to address system performance and efficiency requirements.	PO2,PO3	PSO2,PSO5	An, Ap
CO 2	Apply programming skills to develop solutions for common operating system challenges such as process scheduling, memory allocation, and deadlock detection, fostering a deeper understanding of system-level concepts.	PO1,PO2, P06.	PSO1,PSO2, PS06	U, An, Ap.
CO 3	Evaluate and compare different algorithms and mechanisms through hands-on implementation, enabling informed decision-making in selecting the most suitable approach for specific system requirements.	PO2,PO7	PSO5,PSO7	An, Ap.

CO 4	Acquire practical experience in implementing operating system concepts, enhancing problem-solving abilities and preparing students for real-world applications in software development and system administration roles.	PO2,PO4, PO7,PO8	PSO2,PSO4 ,PSO7	An

List of Experiments:

1. Write a Program to implement First Come First Serve(FCFS) Scheduling.
2. Write a Program to implement Shortest Job First (SJF) Scheduling.
3. Write a Program to implement Priority based Scheduling.
4. Write a Program to implement Round Robin.
5. Write a Program to implement FIFO page replacement algorithm.
6. Write a Program to implement LRU Page replacement algorithm.
7. Write a Program to implement an Optimal page replacement algorithm
8. Write a Program to implement Dinning Philosophers.
9. Write a Program to implement DeadLock Detection.
10. Write a Program to implement the Worst FIT Algorithm.

Textbook(s):

1. Abraham Silberschatz, Peter Galvin, Greg Gagne, Operating System Concepts, Wiley,8th Edition

Additional Reference(s):

1. Achyut S. Godbole, Atul Kahate, Operating Systems, Tata McGraw Hill

2. Naresh Chauhan, Principles of Operating Systems, Oxford Press

3. Andrew S Tanenbaum, Herbert Bos, Modern Operating Systems, 4e Fourth Edition,Pearson Education, 2016

Examination (Total Marks): 50 (Two practical's of 25 marks each)
Experiment Marks: 15
Journal Marks: 05
Viva Marks: 05

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

Course: SY B.Sc. Computer Science
Semester-III Paper-II
Title: Core Java
Course Code: GNKUSCSMJ2103
Credits: 3
No of lectures (Hours): 45
Marks: 100 (75:25)

Course Objectives:

Sr. No.	Course objectives
The course aims at:	
1	Get familiar with basic constructs of programming such as data, operations, conditions, loops, functions etc.
2	The objective of this course is to teach the learner how to use Object Oriented paradigm to develop code and understand the concepts of Core Java.
3	Develop ability for solving problem

Course Outcomes (COs):

Sr. No.	On completing the course, the student will be able to:	POs addressed	PSOs addressed	Cognitive Levels Addressed
CO 1	Design java applications using basic concepts.	PO1,PO2	PSO1,PSO2,PSO6	C,U,Ap
CO 2	Understand the various Object oriented programming concepts using Java	PO1,PO2	PSO1,PSO5, PSO6	U, An, Ap
CO 3	Understand and use string manipulation functions	PO1,PO2,PO6	PSO6	An, Ap
CO 4	Develop and implement exception handling, multithreading.	PO2,PO4	PSO5,PSO6	An, Ap
CO 5	Knowledge and implementation of multithreading	PO1,PO2,PO6	PSO2,PSO5, PSO6	C,U,Ap
CO 6	Develop and implement different I/O handling classes	PO1,PO2,PO3	PSO6	C,U,Ap
CO 7	Understanding Wrapper classes	PO2	PSO5,PSO6	C,U,Ap
CO 8	Understanding collections framework	PO2	PSO5,PSO6	C,U,Ap
CO 9	Knowledge and implementation of the GUI using swing package	PO1,PO2,PO5	PSO5,PSO6	C,U,Ap

Unit		Title	No. of lectures	CO Mapping
Unit 1			**15**	
	1.1	The Java Language: Features of Java, Java programming format, Java Tokens, Java Statements, Java Data Types, Typecasting, Arrays		**CO 1**
	1.2	OOPS: Introduction, Class, Object, Static Keywords, Constructors, this Keyword, Inheritance, super Keyword, Polymorphism (overloading and overriding), Abstraction, Encapsulation, Abstract Classes, Interfaces		**CO 2**
	1.3	Inner Classes: Introduction, Member inner class, Static inner class, Local inner class, Anonymous inner class		**CO 3**
Unit 2			**15**	
	2.1	String Manipulations: String, String Buffer, String Tokenize Packages: Introduction to predefined packages (java.lang, java.util, java.io, java.sql, java.swing), User Defined Packages, Access specifiers		**CO 4**
	2.2	Exception Handling: Introduction, Pre-Defined Exceptions, Try-Catch- Finally,Throws, throw, User Defined Exception examples Multithreading: Thread Creations, Thread Life Cycle, Life Cycle Methods,Synchronization, Wait() notify() notify all() methods		**CO 5**
	2.3	I/O Streams: Introduction, Byte-oriented streams, Character- oriented streams,File, Random access File, Serialization Wrapper Classes: Introduction, Byte, Short, Integer, Long, Float, Double, Character, Boolean classes		**CO 6**
Unit 3			**15**	
	3.1	**Networking:** Network Basics and Socket overview, TCP/IP client sockets, URL, TCP/IP server sockets, Datagrams, java.net package Socket, ServerSocket, InetAddress, URL, URLConnection		**CO 7**

	3.2	Collection Framework: Introduction, util Package interfaces, List, Set, Map, List interface & its classes, Set interface & its classes, Map interface & its classes		**CO 8**
	3.3	**GUI Programming:** Need for swing components, Difference between AWT and swing, Event Handling: The Delegation Event Model, Event classes, Event Listener, Components hierarchy, Panes, Swing components: Jlabel, JTextField and JPasswordField, JTextAres, JButton, JCheckBox, JRadioButton, JComboBox and JList		**C0 9**

Textbook(s):

1) Herbert Schildt, Java The Complete Reference, Ninth Edition, McGraw-Hill Education, 2014

Additional Reference(s):

1) E. Balagurusamy, Programming with Java, Tata McGraw-Hill Education India, 2014
2) Programming in JAVA, 2nd Ed, Sachin Malhotra & Saurabh Choudhary, Oxford Press
3) The Java Tutorials: http://docs.oracle.com/javase/tutorial/

Examination:

- **Internal Examination (25 Marks):** 20 Marks exam (MCQ and short answer question) with 20% completed syllabus. Duration of exam: 40 minutes. And 5 Marks for either Quiz/Assignments /Class Participation etc.
- **End Semester theory examination (75 Marks):** Weightage of each unit will be proportional to the number of lecture hours as mentioned in the syllabus. Duration of exam: 2hours 30mins
- **Combined passing of 40% with minimum 20% in Internal Component.**

Course: SY B.Sc. Computer Science
Practical Semester-III
Course Title: Core Java
Practical Paper -II
Course Code: GNKUSCSMJ2P103
Credits: 01
No of Practical (Hours): 30
Marks: 50

Course Objectives:

Sr. No.	Course objectives
The course aims at:	
1	The objective of this course is to teach the learner how to use Object Oriented paradigm to develop code and understand the concepts of Core Java and to cover-up with the prerequisites of Core java.

Course Outcomes (COs):

Sr. No.	On completing the course, the student will be able to:	POs addressed	PSOs addressed	Cognitive Levels addressed
CO 1	Object oriented programming concepts using Java.	PO1,PO2	PSO1,PSO2,PSO6	C,U,Ap
CO 2	Knowledge of input, its processing and getting suitable output.	PO1,PO2	PSO1,PSO5, PSO6	U, An, Ap
CO 3	Design multithreading concepts	PO1,PO2,PO6	PSO6	An, Ap
CO 4	Understand, design, implement and evaluate IO handling	PO2,PO4	PSO5,PSO6	An, Ap
CO 5	Knowledge and implementation of swing package.	PO1,PO2,PO6	PSO2,PSO5, PSO6	C,U,Ap

List of Experiments:

1. Accept integer values for a, b and c which are coefficients of quadratic equations. Find the solution of the quadratic equation.
2. Accept two n x m matrices. Write a Java program to find addition of these matrices.
3. Accept n strings. Sort names in ascending order.
4. Create a package: Animals. In the package animals create an interface Animal with suitable behaviors. Implement the interface Animal in the same package animals.
5. Demonstrate Java inheritance using extended keywords.
6. Demonstrate method overloading and method overriding in Java.
7. Demonstrate creating your own exception in Java.

8. Using various swing components, design Java applications to accept a student's resume. (Design form)
9. Write a Java List example and demonstrate methods of Java List interface.
10. Design a simple calculator GUI application using swing components.

Textbook(s):

1) Herbert Schildt, Java The Complete Reference, Ninth Edition, McGraw-Hill Education, 2014

Additional Reference(s):

References:

1) E. Balagurusamy, Programming with Java, Tata McGraw-Hill Education India, 2014
2) Programming in JAVA, 2nd Ed, Sachin Malhotra & Saurabh Choudhary, Oxford Press
3) The Java Tutorials: http://docs.oracle.com/javase/tutorial/

Examination (Total Marks): 50
Experiment Marks: 15
Journal Marks: 05
Viva Marks: 05

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

Course: SY B.Sc. Computer Science
Semester-III Paper Minor Course
Title: Web Technologies
Course Code: GNKUSCSMI103
Credits: 3
No of lectures (Hours): 45
Marks: 100 (75:25)

Course Objectives:

Sr. No.	Course objectives
The course aims at:	
1	To provide insight into emerging technologies to design and develop state of the art web applications using client-side scripting, server-side scripting, and database connectivity.
2	On completion of this course, a student will be familiar with client server architecture and able to develop a web application.

Course Outcomes (COs):

Sr. No.	On completing the course, the student will be able to:	POs addressed	PSOs addressed	Cognitive Levels Addressed
CO 1	To design valid, well-formed, scalable, and meaningful pages using emerging technologies.	PO1,PO2	PSO1,PSO2	Ap, An
CO 2	Understand the various interactive designing elements.	PO1,PO2	PSO1,PSO2	U,Ap
CO 3	Explore the basics of CSS and its features.	PO1,PO2	PSO1,PSO2	U,Ap, An
CO 4	To Work with JavaScript and its data types, variables, operators and functions.	PO1,PO2	PSO1,PSO2	Ap, An
CO 5	Design and apply XML to create a markup language for data and document centric applications.	PO1,PO2	PSO1,PSO2	Ap, An
CO 6	Introduce and work with JavaScript objects.	PO1,PO2	PSO1,PSO2	Ap, An
CO 7	To create websites and applications that offer a seamless and dynamic user experience	PO1,PO2	PSO1,PSO2, PSO7	Ap, An,C
CO 8	To use PHP to add some dynamic aspects to web pages	PO1,PO2	PSO1,PSO2, PSO8	Ap, An
CO 9	To Design and build rich interactive web applications.	PO1,PO2	PSO1,PSO2, PSO7	Ap, An,C

Unit		Title	No. of lectures	CO Mapping
Unit 1			**15**	
	1.1	**HTML5:** Fundamental Elements of HTML, Formatting Text in HTML, Organizing Text in HTML, Links and URLs in HTML, Tables in HTML, Images on a Web Page,Image Formats, Image Maps, Colors, FORMs in HTML		**CO 1**
	1.2	**Interactive Elements, Working with Multimedia** - Audio and Video File Formats, HTML elements for inserting Audio / Video on a web page		**CO 2**
	1.3	**CSS:** Understanding the Syntax of CSS, CSS Selectors, Inserting CSS in an HTML Document, CSS properties to work with background of a Page, CSS properties to work with Fonts and Text Styles, CSS properties for positioning an element		**CO 3**
Unit 2			**15**	
	2.1	**JavaScript:** Using JavaScript in an HTML Document, Programming **Fundamentals of JavaScript :** Variables, Operators, Control Flow Statements, Popup Boxes, Functions – Defining and Invoking a Function, Defining Function arguments, Defining a Return Statement, Calling Functions with Timer		**CO 4**
	2.2	**JavaScript Objects :** String, RegExp, Math, Date, **Browser Objects** : Window, Navigator, History, Location, Document, Cookies, Document Object Model,Form Validation using JavaScript		**CO 5**
	2.3	**XML:** Comparing XML with HTML, Advantages and Disadvantages of XML, Structure of an XML Document, XML Entity References, DTD, XSLT: XSLT Elements and Attributes		**CO 6**
Unit 3			**15**	
	3.1	**AJAX:** AJAX Web Application Model, How AJAX Works, XMLHttpRequest Object– Properties and Methods, Handling asynchronous requests using AJAX		**CO 7**
	3.2	**PHP:** Variables and Operators, Program Flow, Arrays, Working with Files and Directories, Working with Databases, Working with Cookies, Sessions and Headers		**CO 8**
	3.3	**Introduction to jQuery:** Fundamentals, Selectors, methods to access HTML attributes, methods for traversing, manipulators, events, effects.		**C0 9**

Text Book(s):

1) HTML 5 Black Book, Covers CSS 3, JavaScript, XML, XHTML, AJAX, PHP and jQuery, 2ed, Dreamtech Press

2) Web Programming and Interactive Technologies, scriptDemics, StarEdu Solutions India.

3) PHP: A Beginners Guide, Vikram Vaswani, TMH

Additional Reference(s):

1) HTML, XHTML, and CSS Bible Fifth Edition, Steven M. Schafer, WILEY

2) Learn to Master HTML 5, scriptDemics, StarEdu Solutions Pvt Ltd.

3) Learning PHP, MySQL, JavaScript, CSS & HTML5, Robin Nixon, O'Reilly

4) PHP, MySQL, JavaScript & HTML5 All-in-one for Dummies, Steve Suehring, Janet Valade Wiley

Examination:

- **Internal Examination (25 Marks):** 20 Marks exam (MCQ and short answer question) with 20% completed syllabus. Duration of exam: 40 minutes. And 5 Marks for either Quiz/Assignments /Class Participation etc.
- **End Semester theory examination (75 Marks):** Weightage of each unit will be proportional to the number of lecture hours as mentioned in the syllabus. Duration of exam: 2hours 30mins
- **Combined passing of 40% with minimum 20% in Internal Component.**

Course: SY B.Sc. Computer Science
Practical Semester-III
Course Title: Practical Minor Course Web Technologies
Practical Paper-III
Code: GNKUSCSMIP103
Credits: 01
No of Practical (Hours): 30
Marks: 50

Course Objectives:

Sr. No.	Course objectives
The course aims at:	
1	To provide insight into emerging technologies to design and develop state of the art web applications using client-side scripting, server-side scripting, and database connectivity.
2	On completion of this course, a student will be familiar with client server architecture and able to develop a web application.

Course Outcomes (COs):

Sr. No.	On completing the course, the student will be able to:	POs addressed	PSOs addressed	Cognitive Levels addressed
CO 1	To design valid, well-formed, scalable, and meaningful pages using emerging technologies.	PO1,PO2	PSO1,PSO2	Ap, An
CO 2	Understand the various platforms, devices, display resolutions, viewports, and browsers that render websites	PO1,PO2	PSO1,PSO2, PSO7	U,Ap, An
CO 3	To develop and implement client-side and server-side scripting language programs.	PO1,PO2	PSO1,PSO2	Ap, An
CO 4	To develop and implement Database Driven Websites.	PO1,PO2	PSO1,PSO2, PSO8	R,Ap,C
CO 5	Design and apply XML to create a markup language for data and document centric applications.	PO1,PO2	PSO1,PSO2	Ap, An

List of Experiments:

1. Design a webpage that makes use of

a. Document Structure Tags b. Various Text Formatting Tags

c. List Tags d. Image and Image Maps

2. Design a webpage that makes use of

a. Table tags b. Form Tags (forms with various form elements)

c. Navigation across multiple pages d. Embedded Multimedia elements

3. Design a webpage that make use of Cascading Style Sheets with

a. CSS properties to change the background of a Page

b. CSS properties to change Fonts and Text Styles

c. CSS properties for positioning an element

4. Write JavaScript code for

a. Performing various mathematical operations such as calculating factorial / finding Fibonacci Series / Displaying Prime Numbers in given range / Evaluating Expressions / Calculating reverse of

a number

b. Validating the various Form Elements

5. Write JavaScript code for

a. Demonstrating different JavaScript Objects such as String, RegExp, Math, Date

b. Demonstrating different JavaScript Objects such as Window, Navigator, History, Location, Document,

c. Storing and Retrieving Cookies

6. Create a XML file with Internal / External DTD and display it using

a. CSS b. XSL

7. Design a webpage to handle asynchronous requests using AJAX on

a. Mouseover b. button click

8. Write PHP scripts for

a. Retrieving data from HTML forms

b. Performing certain mathematical operations such as calculating factorial / finding Fibonacci Series / Displaying Prime Numbers in given range / Evaluating Expressions / Calculating reverse of a number

c. Working with Arrays

d. Working with Files (Reading / Writing)

9. Write PHP scripts for

a. Working with Databases (Storing Records / Reprieving Records and Display them)

b. Storing and Retrieving Cookies

c. Storing and Retrieving Sessions

10. Design a webpage with some jQuery animation effects.

Text Book(s):

1) HTML 5 Black Book, Covers CSS 3, JavaScript, XML, XHTML, AJAX, PHP and jQuery,
2ed, Dreamtech Press

2) Web Programming and Interactive Technologies, scriptDemics, StarEdu Solutions India.

3) PHP: A Beginners Guide, Vikram Vaswani, TMH

Additional Reference(s):

1) HTML, XHTML, and CSS Bible Fifth Edition, Steven M. Schafer, WILEY
2) Learn to Master HTML 5, scriptDemics, StarEdu Solutions Pvt Ltd.
3) Learning PHP, MySQL, JavaScript, CSS & HTML5, Robin Nixon, O'Reilly
4) PHP, MySQL, JavaScript & HTML5 All-in-one for Dummies, Steve Suehring, Janet Valade Wiley

Examination (Total Marks): 50 (Two practical's of 25 marks each)
Experiment Marks: 15
Journal Marks: 05
Viva Marks: 05

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

Course: SY B.Sc. Computer Science
Semester-III Paper-Open Elective (OE)
Course Title: Scientific Programming-I
Course Code: GNKUSCSOE103
Credits: 1
No of lectures (Hours): 30
Marks: 50

Course Objectives:

Sr. No.	Course objectives
The course aims at:	
1	Gain a foundational understanding of computational science principles, including the application of scientific computing techniques to solve complex problems across various domains.
2	Develop proficiency in Python programming, covering fundamental concepts such as data types, control structures, functions, file handling, and error handling, essential for scientific computing tasks.
3	Acquire knowledge and skills in object-oriented programming, including class design, encapsulation, and inheritance, to efficiently organize and manipulate data structures and algorithms.
4	Explore advanced topics in scientific computing, including array computing, curve plotting, and vector/matrix operations using libraries like NumPy, SciPy, and Matplotlib, enabling the manipulation and visualization of scientific data.
5	Enhance capabilities in data analysis and manipulation with Pandas, focusing on data alignment, aggregation, summarization, computation, and analysis, critical for processing and deriving insights from structured data sets in scientific research and applications.

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 will develop a comprehensive understanding of computational science principles and their application in solving real-world scientific problems. They will gain proficiency in utilizing various tools and languages to analyze, model, and simulate complex scientific	PO1,PO2	PSO1,PSO2	Ap, An

	phenomena effectively.			
CO 2	students will possess a thorough understanding of Python programming fundamentals, including data types, control structures, functions. They will be proficient in writing Python scripts to manipulate data, implement algorithms, and handle various computational tasks effectively.	PO1,PO2	PSO1,PSO2, PSO7	U,Ap, An

Unit		**Title**	**No. of lectures**	**CO Mapping**
Unit 1			**15**	
	1.1	Introduction to Computational Science, Applications involving scientific computing, Tools and languages to solve complex scientific problems. Programming in Python- Interpreter and its environment; Introduction to data types, concepts of mutability, operators and variables; conditional and iteration; functions		**CO 1**
Unit II			**15**	
	2.1	Object-oriented programming: classes and methods - encapsulation, inheritance Scientific computation using python - Statistical data analysis, image processing, web development and hardware interfacing using Python		**CO 2**

Textbook(s):

1.Martin C. Brown, Python: The Complete Reference, McGraw Hill Education

Additional Reference(s):

1. Hemant Kumar Mehta, Mastering Python Scientific Computing, Packt Publishing Limited

2. Hans PetterLangtangen, A Primer on Scientific Programming with Python (Link)

3. Claus Fuhrer, Jan Erik Solem, Olivier Verdier, Scientific Computing with Python 3,Packt Publishing Limited

4. Sergio J. Rojas G., Erik A. Christensen, Francisco J. Blanco-Silva, Learning SciPy for Numerical and Scientific Computing, Packt Publishing Limited

Examination:

- **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: SY B.Sc. Computer Science
Practical Semester-III
Course Title: Vocational Skill Course (VSC) PL/SQL Practical
Course Code: GNKUSCSVSC103
Credits: 02
No of Practical (Hours): 60
Marks: 50

Course Objectives:

Sr. No.	Course objectives
The course aims at:	
1	To develop understanding of concepts and techniques for data management and learn about widely used systems for implementation and usage.
2	To develop understanding of Transaction management and crash recovery.
3	To develop concepts of programming concepts of database.

Course Outcomes (COs):

Sr. No.	On completing the course, the student will be able to:	POs addressed	PSOs addressed	Cognitive Levels addressed
CO 1	Analyze and master the concepts of stored procedure, functions, cursors and triggers and its use.	PO1, PO2, PO3	PSO3,PSO4	U, An,E, C
CO 2	Learn about using PL/SQL for data management.	PO3	PSO3	U,An, C

List of Experiments:

1. Writing PL/SQL Blocks with basic programming constructs by including following:
   a. Sequential Statements b. unconstrained loop
2. Sequences: a. Creating simple Sequences with clauses like START WITH, INCREMENT BY, MAXVALUE, MINVALUE, CYCLE | NOCYCLE, CACHE | NOCACHE, ORDER | NOORECER. b. Creating and using Sequences for tables.
3. Writing PL/SQL Blocks with basic programming constructs by including following:
   a. If...then...Else, IF...ELSIF...ELSE... END IF
   b. Case statement
4. Writing PL/SQL Blocks with basic programming constructs for following Iterative Structure:
   a. While-loop Statements
   b. For-loop Statements.
5. Writing PL/SQL Blocks with basic programming constructs by including a GoTO to jump out of a loop and NULL as a statement inside IF.
6. Writing Procedures in PL/SQL Block
   a. Create an empty procedure, replace a procedure and call procedure

b. Create a stored procedure and call it

c. Define procedure to insert data

d. A forward declaration of procedure

7. Writing Functions in PL/SQL Block.

   a. Define and call a function

   b. Define and use function in select clause,

   c. Call function in dbms_output.put_line

   d. Recursive function

   e. Count Employee from a function and return value back

   f. Call function and store the return value to a variable

8. Creating and working with Insert/Update/Delete Trigger using Before/After clause.
9. Write an Implicit and explicit cursor to complete the task.
10. Write a SQL block to handle exception by writing:

    a. Predefined Exceptions,

    b. User-Defined Exceptions,

    c. Redeclared Predefined Exceptions

Textbook:

1. Mastering PL/SQL Through Illustrations: From Learning Fundamentals to Developing Efficient PL/SQL Blocks, Dr. B. Chandra, BPB Publication, 2020
2. Oracle Pl/Sql Training Guide., Training guide, BPB Publications, 2016

Additional References:

1. Abraham Silberschatz, Henry F. Korth,S. Sudarshan , Database System Concepts, 6th Edition 2019

Examination (Total Marks): 50 (25 marks VSC/SEC and 25 marks VSC/SEC subjects)

Experiment Marks: 15

Journal Marks: 05

Viva Marks: 05

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

Course: SY B.Sc. Computer Science
Semester-III: Ability Enhancement Course (AEC)
Course Title: Graph Theory
Course Code: GNKUSCSAEC103
Credits: 2
No of lectures (Hours): 30
Marks: 50 (30:20)

Course Objectives:

Sr. No.	Course objectives
The course aims at:	
1	To give the learner a broad exposure of combinatorial Mathematics through applications especially the Computer Science applications.

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 graph theoretical concepts to understand Computer Science concepts apply them to solve problems	PO1, PO2,PO3	PSO1, PSO2	R, U,Ap,An
CO 2	Understand the basics of graphs and trees	PO2	PSO1,PSO3	R,U, Ap
CO 3	Apply Ramsey methods for a problem.	PO1, PO2	PSO1	R, U,Ap
CO 4	Understand the network flow terminologies	PO1	PSO4	R,U
CO 5	Apply combinatorial view of network flows	PO2,PO3	PSO2	R,U
CO 6	Solve practical and theoretical challenges using graph coloring	PO3,PO4	PSO1,PSO2	U,Ap,An

Unit		Title	No. of lectures	CO Mapping
Unit 1			**15**	
	1.1	**Graph Theory:** Basic Notation and Terminology, Multigraphs: Loops and Multiple Edges, Eulerian and Hamiltonian Graphs		**CO 1**
	1.2	Graph Coloring, Planar Counting, Labeled Trees, A Digression into Complexity Theory.		**CO 2**
	1.3	Small Ramsey Numbers, Estimating Ramsey Numbers, Applying Probability to Ramsey Theory, Ramsey's Theorem The Probabilistic Method		**CO 3**
Unit 2			**15**	
	2.1	**Network Flows:** Basic Notation and Terminology, Flows and Cuts, Augmenting Paths, The Ford-Fulkerson Labeling Algorithm, A Concrete Example, Integer Solutions of Linear Programming Problems.		**CO 4**
	2.2	**Combinatorial Applications of Network Flows:** Introduction,Matching in Bipartite Graphs, Chain partitioning, Pólya's Enumerations		**CO 5**
	2.3	Coloring the Vertices of a Square		**CO 6**

Textbook:

1. Graph Theory: Modeling, Applications and Algorithms, Agnarsson, Pearson Education India (2008).

Additional References:

1. Applied Combinatorics, Mitchel T. Keller and William T. Trotter, 2016, http://www.rellek.net/appcomb.
2. Graph Theory and Combinatorics, Ralph P. Grimaldi, Pearson Education; Fifth edition (2012)

Examination:

- **Internal Examination (20 Marks):** Continuous Internal assessment (CIA) of 20 Marks each (Quiz, Assignment, presentation).
- **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: 1hour.
- **Combined passing of 40% with minimum 20% in Internal Component.**

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

SEMESTER-IV

Course: SY B.Sc. Computer Science
Semester-IV Paper-I
Course Title: Theory of Computation
Course Code: GNKUSCSMJ1104
Credits: 3
No of lectures (Hours): 45
Marks: 100 (75:25)

Course Objectives:

Sr. No.	Course objectives
The course aims at:	
1	To give an overview of the theoretical foundations of computer science from the perspective of formal languages
2	To illustrate finite state machines to solve problems in computing
3	To explain the hierarchy of problems arising in the computer sciences.
4	To familiarize Regular grammars, context free grammar.

Course Outcomes (COs):

Sr. No.	On completing the course, the student will be able to:	POs addressed	PSOs addressed	Cognitive Levels addressed
CO 1	Learn about Automata theory and its application in Language Design	PO1,PO2	PSO1,PSO2	R,U,Ap
CO 2	Learn about Automata theory with outputs	PO1,PO2	PSO1	R,U,An
CO 3	Understand Grammar and Languages	PO2	PSO1,PSO2	U,Ap,An
CO 4	Understand Regular grammar and set	PO1,PO3	PS01	U,Ap,E
CO 5	Simplify CGF and use its applications	PO3	PSO3	Ap,An,E
CO 6	Role of PDA and CFG	PO3	PSO1	Ap,An.E
CO 7	Understand Linear Bound Automata and Languages.	PO2	PSO2	Ap,E
CO 8	Understand Turing machine construction	PO1	PSO1,PSO2	Ap,An
CO 9	Know about unsolvable problems in TM	PO2	PSO2	R,U,Ap

Unit		Title	No. of lectures	CO Mapping
Unit 1			**15**	
	1.1	**Automata Theory:** Defining Automaton, Finite Automaton, Transitions and Its properties, Acceptability by Finite Automaton, Nondeterministic Finite State Machines		**CO 1**
	1.2	DFA and NDFA equivalence, Mealy and Moore Machines, Minimizing Automata.		**CO 2**
	1.3	**Formal Languages:** Defining Grammar, Derivations, Languages generated by Grammar, Chomsky Classification of Grammar and Languages, Recursive Enumerable Sets, Operations on Languages, Languages and Automata		**CO 3**
Unit 2			**15**	
	2.1	**Regular Sets and Regular Grammar:** Regular Grammar, Regular Expressions, Finite automata and Regular Expressions, Pumping Lemma and its Applications, Closure Properties		**CO 4**
	2.2	**Regular Sets and Regular Grammar Context Free Languages:** Context-free Languages, Derivation Tree, Ambiguity of Grammar, CFG simplification, Normal Forms, Pumping Lemma for CFG		**CO 5**
	2.3	**Pushdown Automata:** Definitions, Acceptance by PDA, PDA and CFG		**CO 6**
Unit 3			**15**	
	3.1	**Linear Bound Automata:** The Linear Bound Automata Model, Linear Bound Automata and Languages.		**CO 7**
	3.2	**Turing Machines:** Turing Machine Definition, Representations, Acceptability by Turing Machines, Designing and Description of Turing Machines, Turing Machine Construction, Variants of Turing Machine		**CO 8**

	3.3	**Undecidability:** The Church-Turing thesis, Universal Turing Machine, Halting Problem, Introduction to Unsolvable Problems		**C0 9**

Reference(s):

Textbooks:

1. Theory of Computer Science, K. L. P Mishra, Chandrasekharan, PHI,3rd Edition 2019
2. Introduction to Computer Theory, Daniel Cohen, Wiley, 2nd Edition, 2007
3. Introductory Theory of Computer Science, E.V. Krishnamurthy, Affiliated East-West Press, 2009

Additional References:

1. Theory of Computation, Kavi Mahesh, Wiley India, 2018
2. Elements of The Theory of Computation, Lewis, Papadimitriou, PHI, 2015
3. Introduction to Languages and the Theory of Computation, John E Martin, McGraw-Hill Education, 2010
4. Introduction to Theory of Computation, Michel Sipser, Thomson
5. Introduction to Automata Theory, Languages and Computation, John E. Hopcroft, Pearson Education, 2014

Examination:

- **Internal Examination (25 Marks):** 20 Marks exam (MCQ and short answer question) with 20% completed syllabus. Duration of exam: 40 minutes. And 5 Marks for either Quiz/Assignments /Class Participation etc.
- **End Semester theory examination (75 Marks):** Weightage of each unit will be proportional to the number of lecture hours as mentioned in the syllabus. Duration of exam: 2hours 30mins
- **Combined passing of 40% with minimum 20% in Internal Component.**

Course: SY B.Sc. Computer Science
Practical Semester-IV
Course Title: Theory of Computation
Practical Paper-I
Course Code: GNKUSCSMJ1P104
Credits: 01
No of Practical (Hours): 30
Marks: 50

Course Objectives:

Sr. No.	Course objectives
The course aims at:	
1	To give an overview of the theoretical foundations of computer science from the perspective of formal languages
2	To illustrate finite state machines to solve problems in computing
3	To explain the hierarchy of problems arising in the computer sciences.
4	To familiarize Regular grammars, context free grammar.

Course Outcomes (COs):

Sr. No.	On completing the course, the student will be able to:	POs addressed	PSOs addressed	Cognitive Levels addressed
CO 1	Program the automata theory, PDA concepts	PO1,PO2	PSO1,PSO2	C
CO2	Design a turing machine concepts	PO1,PO2	PSO1,PSO2	C

List of Experiments:

1. Write a program for tokenization of given input
2. Write a program for generating regular expressions for regular grammar
3. Write a program for generating derivation sequence / language for the given sequence of productions
4. Design a Program for creating machines that accepts three consecutive ones.
5. Design a Program for creating a machine that accepts the string always ending with 101.
6. Design a program for accepting decimal numbers divisible by 2.
7. Design a program for creating a machine which accepts string having equal no. of 1's and 0's.
8. Design a program for creating a machine which count number of 1's and 0's in a given string.
9. Design a PDA to accept WCWR where w is any string and WR is the reverse of that string and C is a special symbol.
10. Design a Turing machine that's accepts the following language an b n c n where n>0

Textbook:

1. Theory of Computer Science, K. L. P Mishra, Chandrasekharan, PHI,3rd Edition 2019
2. Introduction to Computer Theory, Daniel Cohen, Wiley, 2nd Edition, 2007

Additional References:

1. Introductory Theory of Computer Science, E.V. Krishnamurthy, Affiliated East-West Press, 2009
2. Theory of Computation, Kavi Mahesh, Wiley India, 2018
3. Elements of The Theory of Computation, Lewis, Papadimitriou, PHI, 2015
4. Introduction to Languages and the Theory of Computation, John E Martin, McGraw-Hill Education, 2010 4. Introduction to Theory of Computation, Michel Sipser, Thomson
5. Introduction to Automata Theory, Languages and Computation, John E. Hopcroft, Pearson Education, 2014

Examination (Total Marks): 50 (Two practical's of 25 marks each)
Experiment Marks: 15
Journal Marks: 05
Viva Marks: 05

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

Course: SY B.Sc. Computer Science
Semester-IV Paper-II
Course Title: Web Programming with Java
Course Code: GNKUSCSMJ2104
Credits: 3
No of lectures (Hours): 45
Marks: 100 (75:25)

Course Objectives:

Sr. No.	Course objectives
The course aims at:	
1	Develop and Explore advanced topics of Java web programming.
2	Understand the concepts related to Java database connectivity
3	Explore and understand use of Servlet Programming
4	Explore and understand use of Java Server Programming
5	To Understand Web Services and implementation model for SOA
6	Explore and understand Hibernate & spring framework

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 Database connectivity in Java for given application.	PO1 PO2	**PSO 8**	C, Ap, R
CO 2	Understanding JSON data format	PO1 PO2	**PSO 2**	U
CO 3	Implement webpage with dynamic content and server side web application using Servlet.	PO1 PO2	**PSO 5** **PSO 6** **PSO 8**	Ap, U
CO 4	Implement webpage with dynamic content and server side web application using JSP.	PO1 PO2 PO7	**PSO 5** **PSO 6** **PSO 8**	C,Ap,U
CO 5	Efficiently use market leading environment tools to create and consume web services	PO1 PO2	**PSO 5** **PSO 6**	C,Ap,U

			PSO 8	
CO 6	Apply OOP principles to creation of webservice solutions.	PO1 PO2	**PSO 5** **PSO 6** **PSO 8**	C,Ap,U
CO 7	Use Object Relation Mapping using Hibernate to build database dependent applications	PO1 PO2	**PSO1** **PSO 5** **PSO 6** **PSO 8**	C,Ap,U
CO 8	Understand core concept of Spring Framework	PO1 PO2	PSO 1 PSO 2	C,Ap,U
CO 9	Understand & implement AOP concept in Spring	PO1 PO2 PO7	**PSO 6**	C,Ap,U

Unit		**Title**	**No. of lectures**	**CO Mapping**
Unit 1			**15**	
	1.1	**Database Programming using JDBC:** Introduction, JDBC Architecture, Types of Drivers, Statement, ResultSet, Read Only ResultSet, Updatable ResultSet, Forward Only ResultSet, Scrollable ResultSet, PreparedStatement, Connection Modes, SavePoint, Batch Updations, CallableStatement, BLOB & CLOB		**CO 1**
	1.2	**JSON:** Overview, Syntax, DataTypes, Objects, Schema, Comparison with XML, JSON with Java		**CO 2**
	1.3	**Servlets:** Introduction, Web application Architecture, Http Protocol & Http Methods, Web Server & Web Container, Servlet Interface, GenericServlet, HttpServlet, Servlet Life Cycle		**CO 3**
Unit 2			**15**	
	2.1	ServletConfig, ServletContext, Servlet Communication, Session Tracking Mechanisms		**CO 4**
	2.2	**JSP:** Introduction, JSP LifeCycle, JSP Implicit Objects & Scopes, JSP Directives, JSP Scripting Elements, JSP Actions: Standard actions and customized actions, Introduction to JSP Standard Tag Library (JSTL) and JSTL Tags.		**CO 5**

	2.3	**Web Service Architecture** – Web services Architecture and its characteristics, core building blocks of web services, standards and technologies available for implementing web services, web services communication, basic steps of implementing web services.		**CO 6**
Unit 3			**15**	
	3.1	**Hibernate:** Introduction, Writing the application, application development approach, creating database and tables in MySQL, creating a web application, Adding the required library files, creating a java bean class, creating hibernate configuration and mapping file, adding a mapping resource, creating JSPs.		**CO 7**
	3.2	**Spring Frameworks:** What is Spring, How Spring fits into the Enterprise world, Spring Modules, Spring Core Concepts-What is a Core Container, Introduction to IOC, Types of DI, Setter VS Constructor, Collection DI, Bean Inheritance, Collection Merging, Inner Beans, Bean Aliasing, Bean Scopes, Inner Beans, Null String, Bean Auto wiring		**CO 8**
	3.3	**Spring AOP:** AOP Concepts, Programmatic VS Declarative AOP, Programmatic AOP, Declarative AOP		**C0 9**

Reference(s):

1. Black Book " Java server programming" J2EE, 1st ed., Dream Tech Publishers, 2008.
2. Complete Reference J2EE by James Keogh mcgraw publication
3. Professional Java Server Programming by Subrahmanyam Allamaraju, Cedric Buest Wiley Publication
4. Core Java, Volume II: Advanced Features by Cay Horstmann and Gary Cornell Pearson Publication
5. Java Persistence with Hibernate by Christian Bauer, Gavin King
6. Spring in Action 3rd edition , Craig walls, Manning Publication
7. JDBC™ API Tutorial and Reference, Third Edition, Maydene Fisher, Jon Ellis, Jonathan Bruce, Addison Wesley
8. Web Services & SOA Principles and Technology, Second Edition, Michael P. Papazoglou.

Examination:

- **Internal Examination (25 Marks):** 20 Marks exam (MCQ and short answer question) with 20% completed syllabus. Duration of exam: 40 minutes. And 5 Marks for either Quiz/Assignments /Class Participation etc.
- **End Semester theory examination (75 Marks):** Weightage of each unit will be proportional to the number of lecture hours as mentioned in the syllabus. Duration of exam: 2hours 30mins
- **Combined passing of 40% with minimum 20% in Internal Component.**

Course: SY B.Sc. Computer Science
Practical Semester-IV
Course Title: Web Programming with Java
Practical Paper-II
Course Code: GNKUSCSMJ2P104
Credits: 01
No of Practical (Hours): 30
Marks: 50

Course Objectives:

Sr. No.	Course objectives
The course aims at:	
1	To understand database programming in Java.
2	Learn how to design and program web applications using java.
3	To understand components of Java web programming Program
4	To Understand Web Services and implementation model for SOA
5	Explore and understand Hibernate & spring framework

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 Database connectivity in Java for given application.	PO1 PO2	**PSO 8**	C, Ap, R
CO 2	Understanding JSON data format	PO1 PO2	**PSO 2**	U
CO 3	Implement webpage with dynamic content and server side web application using Servlet.	PO1 PO2	**PSO 5** **PSO 6** **PSO 8**	Ap, U
CO 4	Implement webpage with dynamic content and server side web application using JSP.	PO1 PO2 PO7	**PSO 5** **PSO 6** **PSO 8**	C,Ap,U
CO 5	Efficiently use market leading environment tools to create and consume web services	PO1 PO2	**PSO 5** **PSO 6** **PSO 8**	C,Ap,U

List of Experiments:

1. Practical on database programming using JDBC
2. Develop a Java application to store image in a database as well as retrieve image from database.
3. Practical on JSON
4. Web application using Servlet
5. Session Handling using Servlet
6. Web Application using JSP
7. Web Services creation and usage
8. Practical on Hibernet
9. Develop a Hibernate application to store Customers record in MySQL Database.
10.. Practical on Spring Framework

Reference(s):

1. Black Book " Java server programming" J2EE, 1st ed., Dream Tech Publishers, 2008. 3. Kathy walrath"
2. Complete Reference J2EE by James Keogh mcgraw publication
3. Professional Java Server Programming by Subrahmanyam Allamaraju, Cedric Buest Wiley Publication
4. Core Java, Volume II: Advanced Features by Cay Horstmann and Gary Cornell Pearson Publication
5. Java Persistence with Hibernate by Christian Bauer, Gavin King
6. Spring in Action 3rd edition , Craig walls, Manning Publication
7. JDBC™ API Tutorial and Reference, Third Edition, Maydene Fisher, Jon Ellis, Jonathan Bruce, Addison Wesley

Examination (Total Marks): 50

Experiment Marks: 15

Journal Marks: 05

Viva Marks: 05

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

Course: SY B.Sc. Computer Science
Semester-IV Paper Minor Course Title: Python Programming I
Course Code: GNKUSCSMI104
Credits: 3
No of lectures (Hours): 45
Marks: 100 (75:25)

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	PO1	PSO1,PSO2,	Ap, An,C

	Python	PO2 PO7	PSO5	
CO 9	Ability to create and use dictionaries in Python	PO1 PO2 PO7	PSO1,PSO2	Ap, An,C

Unit		**Title** **Python Programming I**	**No. of lectures**	**CO Mapping**
Unit 1			**15**	
	1.1	**Overview of Python:** History & Versions, Features of Python, Execution of a Python Program, Python Interpreter, Memory Management in Python, Garbage Collection in Python, Comparison of Python with C and Java, Installing Python, Writing and Executing First Python Program, Getting Help, IDLE		**CO 1**
	1.2	**Data Types, Variables and Other Basic Elements:** Comments,Docstrings, Data types- Numeric Data type, Compound Data Type,Boolean Data type, Dictionary, Sets, Mapping, Basic Elements of Python,Variables **Input and Output Operations:** Input Function, Output Statements, The print() function, The print("string") function, The print(variables list) function, , The print(formatted string) function, Command Line Arguments		**CO 2**
	1.3	**Operators:** Arithmetic operators, Assignment operators, Unary minus operator, Relational operators, Logical operators, Bitwise operators, Membership operators, Identity operators, Precedence of Operators,Associativity of Operators		**CO 3**
Unit 2			**15**	
	2.1	**Control Statements:** The if statement, The if ... else Statement, The 'if ...elif ... else' Statement, Loop Statement- while loop, for loop, Infinite loop,Nested loop, The else suite, break statement, continue statement, pass statement, assert statement, return statement		**CO 4**
	2.2	**Arrays:** Creating Arrays, Indexing and Slicing of Arrays, Basic Array Operations, Arrays Processing, Mathematical Operations on Array		**CO 5**

		Modules: Introduction to Modules in Python , Built in Modules, User defined module		
	2.3	**Functions:** Function definition and call, Returning Results, Returning Multiple Values from a Function, Built-in Functions, Parameters and Arguments, Formal and Actual Arguments, Positional Arguments, Keyword Arguments, Default Arguments, Arbitrary Arguments, Recursive Functions, Anonymous or Lambda Functions		**CO 6**
Unit 3			**15**	
	3.1	**Strings:** Creating Strings, Working with Strings, String Handling Functions: Length of a String, Indexing and Slicing, Repeating and Concatenating Strings, Comparing Strings, Removing Spaces, Finding Substrings, Counting Substrings, Sorting Strings, Searching in the Strings, Formatting Strings, Finding the Number of Characters and Words, Inserting Substrings into a String		**CO 7**
	3.2	**List and Tuples:** Lists, List Functions and Methods, List Operations, List Slices, Nested Lists, Tuples, Functions in Tuple		**CO 8**
	3.3	**Dictionaries:** Creating a Dictionary, Operators in Dictionary, Dictionary Methods, Using for Loop with Dictionaries, Operations on Dictionaries, Converting Lists into Dictionary, Converting Strings into Dictionary, Passing Dictionaries to Functions, Sorting the Elements of a Dictionary using Lambda, Ordered Dictionaries		**C0 9**

Reference(s):

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

Examination:

- **Internal Examination (25 Marks):** 20 Marks exam (MCQ and short answer question) with 20% completed syllabus. Duration of exam: 40 minutes. And 5 Marks for either Quiz/Assignments /Class Participation etc.
- **End Semester theory examination (75 Marks):** Weightage of each unit will be proportional to the number of lecture hours as mentioned in the syllabus. Duration of exam: 2hours 30mins
- **Combined passing of 40% with minimum 20% in Internal Component.**

Course: SY B.Sc. Computer Science
Practical Semester-IV
Course Title: Python Programming I
Practical Paper-III
Course Code: GNKUSCSMIP104
Credits: 01
No of Practical (Hours): 30
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 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	Students should be able to understand the concepts of programming	PO1 PO2	PSO1,PSO2	Ap, An
CO 2	Students should be able to develop logic for Problem Solving.	PO1 PO2	PSO1,PSO2	U,Ap
CO 3	Students should be made familiar about the basic constructs of programming such as data, operations, conditions, loops, functions etc.	PO1 PO2	PSO1,PSO2, PSO5	U,Ap, An
CO 4	Students should be able to apply the problem solving skills using syntactically simple language i.e. Python (version: 3.X or higher)	PO1 PO2 PO7	PSO1,PSO2, PSO7	Ap, An
CO 5	Students should be able to Use functions and represent Compound data using Lists, Tuples and Dictionaries	PO1 PO2	PSO1,PSO2	Ap, An,U

List of Experiments:

1. Write a program to design and develop python program to implement various control statement using suitable examples
2. Write a program in Python to define and call functions for suitable problems.
3. Write a Python program to demonstrate different types of function arguments.
4. Write a Python program to demonstrate the precedence and associativity of operators.
5. Write a suitable Python program to implement recursion for problems such as Fibonacci series,

Factorial, Tower of Hanoi etc.

6. Write Python program to implement and use lambda function in python
7. Write a python program to create and manipulate arrays in Python. Also demonstrate use of slicing and indexing for accessing elements from the array.
8. Write a program to implement a list in Python for suitable problems. Demonstrate various operations on it.
9. Write a program to implement tuples in Python for suitable problems. Demonstrate various operations on it.
10. Write a program to implement a dictionary in Python for suitable problems. Demonstrate various operations on it.

Reference(s):

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

Examination (Total Marks): 50 (Two practical's of 25 marks each)
Experiment Marks: 15
Journal Marks: 05
Viva Marks: 05

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

Course: SY B.Sc. Computer Science
Semester-IV Paper-Open Elective (OE)
Course Title: Scientific Programming-II
Course Code: GNKUSCSOE104
Credits: 1
No of lectures (Hours): 30
Marks: 50

Course Objectives:

Sr. No.	Course objectives
The course aims at:	
1	Master the basics of R programming language, including understanding its interpreter, major data structures (vectors, matrices, arrays, lists, and data frames), and control structures for efficient data manipulation and analysis.
2	Gain proficiency in managing packages in R, from installation to usage, for tasks such as reading/writing data files, extracting data from websites, cleaning data, and transforming data through sorting and column manipulation.
3	Develop skills in preprocessing data by centering, scaling, and normalizing values, along with converting data types and utilizing built-in string functions for effective data preparation.
4	Acquire knowledge and techniques for statistical analysis and data visualization, including summarizing data for insights, and creating various plots such as scatter plots, line plots, bar charts, histograms, and box plots, enabling comprehensive exploration and communication of data patterns and trends.

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 will gain proficiency in using R for data manipulation and analysis, mastering fundamental data structures and control structures to efficiently manage and analyze datasets. They will develop skills in creating and implementing functions for streamlined code development and automation in data analysis tasks.	PO1 PO2	PSO1,PSO3	Ap, An
CO 2	Students will become adept at leveraging R packages for diverse tasks such as data import/export, web scraping, and data cleaning,	PO1 PO2	PSO1,PSO2	U,Ap

	enabling them to efficiently handle real-world data sources. They will develop expertise in data transformation techniques including sorting, adding, and removing columns, enhancing their ability to manipulate and prepare data for analysis.			

Unit		Title	No. of lectures	CO Mapping
Unit 1			**15**	
	1.1	Introduction: R interpreter, Introduction to major R data structures like vectors, matrices, arrays, list and data frames, Control Structures, vectorized if and multiple selection, functions. Installing, loading and using packages		**CO 1**
Unit 2			**15**	
	2.1	Read/write data from/in files, extracting data from web-sites, Clean data, Transform data by sorting, adding/removing new/existing columns. Statistical analysis of data - for summarizing and understanding data Visualizing data using scatter plot, line plot, bar chart, histogram and box plot.		**CO 2**

Textbook:

1. Cotton, R., Learning R: a step by step function guide to data analysis. 1st edition. O'reilly Media Inc

Additional References:

1. Gardener, M.(2017). Beginning R: The statistical programming language, WILEY Lawrence, M., & Verzani, J. (2016).
2. Programming Graphical User Interfaces in R. CRC press. (ebook)

Examination:

- **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 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: SY B.Sc. Computer Science
Practical Semester-IV
Course Name: Skill Enhancement Course (SEC)
Course Title: IoT Technology
Practical Paper-4
Course Code: GNKUSCSSEC104
Credits: 02
No of Practical (Hours): 60
Marks: 50

Course Objectives:

Sr. No.	Course objectives
The course aims at:	
1	The course will provide a comprehensive understanding of embedded systems, microcontrollers, and single-board computers.
2	The learner will understand the fundamentals of the smallest computer and its working principles.
3	Gain expertise in Linux commands and effectively explore the Raspbian operating system.
4	The course will also provide practical skills in programming, hardware interfacing, and IoT applications.

Course Outcomes (COs):

Sr. No.	On completing the course, the student will be able to:	POs addressed	PSOs addressed	Cognitive Levels addressed
CO 1	Exhibit knowledge of Arduino Uno and its pins, and successfully interface with the Integrated	PO1, PO2, PO8	PSO1, PSO2, PSO3, PSO4	R, U, Ap

	Development Environment (IDE).			
CO 2	**Develop Python programming skills and write code for execution on Raspberry Pi.**	**PO2, PO3, PO6**	**PSO4, PSO5, PSO6, PSO7**	**U, An, E, C**
CO 3	**Apply GPIO concepts to control an external device using Python on Arduino Uno or Raspberry Pi.**	**PO2, PO4, PO5, PO7, PO8**	**PSO5, PSO6, PSO9, PSO10**	**U, Ap, An, E, C**
CO 4	**Integrate LED with Internet of Things (IoT) using Node RED.**	**PO1, PO2, PO4, PO7**	**PSO3, PSO7, PSO8, PSO9**	**C, Ap, An, E**
CO 5	**Understand and employ a stack of Raspberry Pi for enhanced computing and analysis.**	**PO4, PO6, PO7, PO8**	**PSO1, PSO2, PSO7, PSO9, PSO10**	**U, Ap, An, E, C**

List of Experiments:

1. Understanding the smallest computer and it's working.
2. Discovering different operating systems of Raspberry Pi and Arduino.
3. Preparing Raspberry Pi: Hardware preparation and Installation.
4. Demonstrate Arduino Uno and its pins interfacing with IDE.
5. Linux Commands: Exploring the Raspbian.
6. Writing a Python code and executing it on Raspberry Pi.
7. GPIO: Light the LED with Python with/without a button using either Uno/Raspberry Pi.
8. PIO: LED Grid Module: Program the 8X8 Grid with Different Formulas
9. SPI: Camera Connection and capturing Images using SPI
10. Real Time Clock display using PWM.
11. Stepper Motor Control: PWM to manage stepper motor speed using Uno/Raspberry Pi.
12. Node RED: Connect LED to Internet of Things
13. Use different types of sensors (LDR, Temperature) with Raspberry Pi/Uno
14. Stack of Raspberry Pi for better Computing and analysis
15. Create a simple Web server using Raspberry Pi.

Reference(s):

1. Mastering the Raspberry Pi, Warren Gay, Apress(2014)

2. Introduction to IoT Paperback by Sudip Misra , Anandarup Mukherjee , Arijit Roy , Cambridge Press, 2022

3. Jain, Prof. Satish, Singh, Shashi, "Internet of Things and its Applications", 1st Edition, BPB,

2020.

4. Shriram K Vasudevan, Abhishek S Nagarajan, RMD Sundaram, Internet of Things, Wiley, India, 2019

5. IoT and Edge Computing for Architects - Second Edition, by Perry Lea, Publisher: Packt Publishing, 2020

6. Learning Internet of Things, Peter Waher, Packt Publishing(2015)

Additional Reference(s):

1. Internet of Things by Vinayak Shinde, SYBGEN Learning India Pvt. Ltd, 2020
2. Internet of things, Dr. Kamlesh Lakhwani, Dr. Hemant kumar Gianey, Josef Kofi Wireko, Kamalkant Hiran, BPB Publication, 2020
3. Arduino, Raspberry Pi, NodeMCU Simple projects in easy way by Anbazhagan k and Ambika Parameswari k, 2019.
4. IoT based Projects: Realization with Raspberry Pi, NodeMCU Paperback – February 2020, by Rajesh Singh Anita Gehlot, 2020
5. Abusing the Internet of Things, Nitesh Dhanjani, O'Reilly

Examination (Total Marks): 50 (Two practical's of 20 marks each)

Journal Marks: 05

Viva Marks: 05

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

Course: SY B.Sc. Computer Science
Semester-III: Ability Enhancement Course (AEC)
Course Title: Software Engineering
Course Code: GNKUSCSAEC104
Credits: 2
No of lectures (Hours): 30
Marks: 50 (30:20)

Course Objectives:

Sr. No.	Course objectives
The course aims at:	
1	To learn and understand the Concepts of Software Engineering
2	To learn and understand Software Development Life Cycle
3	To apply the project management, analysis and testing principles to software project development.

Course Outcomes (COs):

Sr. No.	On completing the course, the student will be able to:	POs addressed	PSOs addressed	Cognitive Levels addressed
CO 1	Plan a software engineering process life cycle, including the specification, design, implementation, and testing of software systems that meet specification, performance, maintenance and quality requirements.	PO1,PO2	PSO1,PSO2	R,U,AP, C
CO 2	Analyze the product with different models	PO2	PSO3,PSO4	R,U,An,E
CO 3	Know how to develop the code from the design and effectively apply relevant standards and perform testing, and quality management and practice	PO2,PO3	PSO2	U, Ap, An,E
CO 4	List and manage the risk factors associated with a software product	PO3	PSO3	Ap,An,E
CO 5	Use modern engineering tools necessary for software project management, time management and software reuse.	PO3	PSO4	An,E
CO 6	learn relevant standards and perform testing	PO2,PO3	PSO1.PSO2	Ap,E, C

Unit		Title	No. of lectures	CO Mapping
Unit 1			**15**	
	1.1	**Introduction:** The Nature of Software, Software Engineering, Prescriptive Models: Waterfall Model, Incremental, RAD		**CO 1**
	1.2	**Models Evolutionary Process Models:** Prototyping, Spiral and Concurrent Development Model Specialized Models: Component based, Aspect Oriented development, The Unified Process Phases		**CO 2**
	1.3	**Agile Development:** Agility, Agile Process, Extreme Programming **Requirement Analysis and System Modeling:** Requirements Engineering, Eliciting Requirements, SRS Validation, Components of SRS, Characteristics of SRS **Object-oriented design using the UML -** Class diagram, Object diagram, Use case diagram, Sequence diagram, Collaboration diagram, State chart diagram, Activity diagram, Component diagram, Deployment diagram		**CO 3**
Unit 2			**15**	
	2.1	**Risk Management** - Risk strategies, Software risks, Risk Identification, projection, RMMM Quality Concepts		**CO 4**
	2.2	**Software Quality Assurance:** SQA activities, Software reviews, FTR, Software reliability and measures, SQA plan Software Configuration Management, elements of SCM, SCM Process, Change Control Capability Maturity Model		**CO 5**
	2.3	**Software Testing :** Verification and Validation,Introduction to Testing, Testing Principles, Testing Objectives, Test Oracles, Levels of Testing, White-Box Testing/Structural Testing, Functional/Black-Box Testing, Test Plan, Test-Case Design		**CO 6**

Textbook(s):

1. Software Engineering, A Practitioner's Approach, Roger S, Pressman, 2019
2. Software Engineering: principles and Practices, Deepak Jain, OXFORD University Press, 2008

Additional Reference(s):

1. Software Engineering, Ian Sommerville, Pearson Education, 2017
2. Fundamentals of Software Engineering, Fourth Edition, Rajib Mall, PHI, 2018
3. Software Engineering: Principles and Practices, Hans Van Vliet, John Wiley & Sons, 2010
4. A Concise Introduction to Software Engineering, Pankaj Jalote, Springer

Examination:

- **Internal Examination (20 Marks):** Continuous Internal assessment (CIA) of 20 Marks each (Quiz, Assignment, presentation).
- **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: 1hour.
- **Combined passing of 40% with minimum 20% in Internal Component.**

Course Outcomes

Bloom's Taxonomy categorizes educational objectives into six cognitive levels, listed from the simplest to the most complex.

The course outcomes (CO) are mapped to the revised Bloom's Taxonomy using the following Cognitive levels along with abbreviations: **Remembering-R, Understanding-U, Applying-Ap, Analyzing-An, Evaluating-E, Creating-C.**

R-Remembering: This level involves recalling facts, terms, basic concepts, and answers without necessarily understanding the meaning or implications.

U-Understanding: Understanding involves explaining ideas or concepts and interpreting them in one's own words to demonstrate comprehension.

Ap-Applying: Applying knowledge involves using acquired knowledge in new situations or applying it in different ways.

An-Analyzing: Analyzing involves breaking down information into parts to understand its organizational structure, recognizing patterns, and identifying relationships between components.

E-Evaluating: Evaluating involves making judgments based on criteria and standards, assessing the value of theories, presentations, or materials.

C-Creating: Creating involves putting elements together to form a coherent or functional whole, reorganizing elements into a new pattern or structure.

Use the following action verbs for Blooms taxonomy levels to prepare Course outcomes:

No	Levels	Action verbs
1	**Remember**	Choose, Describe, Define, Label List, Locate,, Match, Memorize, Name, Omit, Recite, Select, State, Count, Draw, Outline, Point, Quote, Recall, Recognize, Repeat, Reproduce, Recall, Arrange, Duplicate, Tabulate.
2	**Understand**	Restate ,Discuss, Clarify, Locate, Recognise, Classify, Translate, Explain, Express, Review, Interpret, Select, Summarise, Contrast, Predict, Associate, Estimate, Extend
3	**Apply**	Demonstrate, Schedule, Operate, Dramatize, Apply, Employ, Use, Practise, Illustrate, Choose, Solve, Write, Calculate, Complete, Show, Examine, Modify, Relate, Classify, Experiment.
4	**Analyse**	Distinguish, Differentiate, Investigate, Categorise, Appraise, Inspect, Test, Debate, Compare, Contrast, Question
5	**Evaluate**	Judge, Score, Select, Evaluate, Choose, Rate, Assess, Compare, Estimate, Value, Measure, Discriminate, Argue, Defend, Support, Conclude, Summarize, Appraise, Revise.
6	**Create**	Compose, Assemble, Organise, Plan, Collect, Propose, Construct, Design, Create, Formulate, Arrange, Devise, Modify, Derive, Develop, Integrate, Rearrange, Substitute, Invent, Generalise.

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

PROGRAMME OUTCOMES (PO)

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 Specific Outcomes (PSOs) for B.Sc. in Computer Science

Sr. No.	A student completing B.Sc. in Computer Science 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.