ST. ALOYSIUS COLLEGE(AUTONOMOUS), JABALPUR

Reaccredited 'A++ 'Grade by NAAC(CGPA:3.58/4.00) College with Potential for Excellence by UGC

DST-FIST Supported & STAR College Scheme by DBT

PART A: Introduction			
PROGRAM: Under Graduate	CLASS: B.Sc.	SEMESTER: I	SESSION: 2025-
Subject: Computer Science			

1.	Course Code	C-1 (TH)
2.	Course Title	Computer System Architecture
3.	Course Type	Core Course
4.	Pre-Requisite (if any)	To study this course, a student must have had the subject Physics/ Math in 12th class.
5.	Course Learning Outcomes(CO)	On completion of this course, learners will be able to: CO1. Understand the basic structure, operation and characteristics of digital computer. CO2. Be able to design simple combinational digital circuits based on given parameters. CO3. Familiarity with working of arithmetic and logic unit as well as the concept of pipelining. CO4. Know about hierarchical memory system including cache memories and virtual memory. CO5. Understand concept and advantages of parallel ism, threading, multi-processors and multi-core processors. Know the contributions of Indians in the field of computer architecture and related technologies.
6.	Credit Value	Theory 4 Credits
7.	Total Marks	Max. Marks : 100 | Min. Passing Marks: 35

PART B: Content of the Course

No. of Lectures (in hours per week): 4 Hrs. per week

Total No. of Lectures: 60 Hrs.

Module	Topics	No. of
I	Fundamentals of Digital Electronics: Number System-Binary, Decimal, Octal, Hexa-Decimal, Conversions, Binary Arithmetic-Addition, Subtraction, Multiplication, Division, Underflow, Overflow, Sign Magnitude, Complements-1's and 2's, Fixed-Point Representation, Floating-Point Representation.	12
II	Boolean Algebra, Reducing Boolean Expression, Logic Gates-AND, OR, NOT, Universal Gates-NAND, NOR, Analog and Digital Signals, Clock Waveform Timing, Map Simplification, K-Map- Two, Three and Four variables.	08
III	Combinational Circuits- Adder, Subtractor, Multiplexer, De-multiplexer, Decoders, Encoders. Binary Codes – Gray Codes, ASCII code, BCD code, EBCDIC, Error Detection Code and Correction Code, Hamming Code.	10

IV	Sequential Circuits - Flip - Flops, SR, D, T, JK, Master-Slave, Registers, Shift Registers- SISO, SIPO, PISO, PIPO, Counters, Instruction, Instruction Format, Instruction Codes, instructions Cycles, Addressing Modes.	10
V	Handshaking, Concepts of RISC, CISC, DMA Data Transfer, Auxiliary Memory, Cache Memory, Associative Memory, Virtual Memory, Flynn's classification - Introduction to SISD, SIMD, MISD, MIMD, Parallelism, Multicore processors.	10
Keywords/Tags: Digital Electronics, Logic Gates, Circuits, Instruction formats, Parallelism, Memory hierarchy, Multicore, Multi-threading, SISD, SIMD, MISD, MIMD.		

CO		PO	PSO	Cognitive Level
CO1	Understand the basic structure, operation and characteristics of digital computer.	PO3	PSO-1	Understanding and Remembering
CO2	Be able to design simple combinational digital circuits based on given parameters.	PO4	PSO-3	Applying and Analysing
CO3	Familiarity with working of arithmetic and logic unit as well as the concept of pipelining.	PO5	PSO-5	Creating, Analysing and Evaluating
CO4	Know about hierarchical memory system including cache memories and virtual memory.	PO4	PSO-3	Creating, Analysing and Evaluating
CO5	Understand concept and advantages of parallel ism, threading, multi-processors and multi-core processors.	PO5	PSO-5	Understanding and Analysing

PART C: Content of the Course		
No. of Lab. Practical s (in hours per week): 2 Hrs. per week		
Total No. of Labs:		
	Suggestive list of Practical	No. of
	1. To study basic gates (AND, OR, NOT) and verify their truth tables. 2. To study and verify NAND as Universal gate using IC 7400. 3. To realize basic gate AND from Universal gate NAND. 4. To realize basic gate OR from Universal gate NAND. 5. To realize basic gate N OT from Universal gate NAND. 6. To study and verify NOR as Universal gate 7. To realize basic gate AND from Universal gate NOR. 8. To realize basic gate OR from Universal gate NOR. 9. To realize basic gate NOT from Universal gate NOR. 10. To study Half Adder using basic gates and verify its truth table. 11. To study Full Adder using basic gates and verify its truth table. 12. To design and construct RS flip Flop using gates and verifies the truth table. 13. To design and construct JK Flip Flop using gates and verifies the truth table. 14. To verify De-Morgan's First Law Theorem. 15. To verify De-Morgan's Second Law Theorem.	
	Keywords/Tags: Digital Electronics, Logic Gates, AN D, OR, NOT, IC7486, IC 7400, NAND, NOR, IC 7483, Circuits, Flip Flop, De-Morgan's Theorem.	

Assessment and Evalution		
Suggested Continuous Evalution Methods: Maximum Marks: 100 Continuous Comprehensive Evaluation (CCE): 30 Marks University Exam (UE): 70 Marks		
Internal Assessment: Continuous Comprehensive Evaluation (CCE)	Class Test Assignment/Presentation	Total Marks: 30
External Assessment: University Exam (UE) Time: 03.00 Hours	Section (A): Objective Type Section (B): Short Questions Section (C): Long Questions	Total Marks: 70

ST. ALOYSIUS COLLEGE(AUTONOMOUS), JABALPUR

Reaccredited 'A++ 'Grade by NAAC(CGPA:3.58/4.00) College with Potential for Excellence by UGC

DST-FIST Supported & STAR College Scheme by DBT

ST. ALOYSIUS COLLEGE(AUTONOMOUS), JABALPUR

Reaccredited 'A++ 'Grade by NAAC(CGPA:3.58/4.00) College with Potential for Excellence by UGC

DST-FIST Supported & STAR College Scheme by DBT

PART A: Introduction		
PROGRAM: Under Graduate I CLASS: B.Sc. I SEMESTER: I I SESSION: 2025-26		
Subject: Computer Science		
1.	Course Code	m-I (TH)
2.	Course Title	Computer Organization
3.	Course Type	Minor Course
4.	Pre-Requisite (if any)	To study this course, Mathematics of 12th standard is desirable.
5.	Course Learning Outcomes(CLO)	On completion of this course, learners will be able to: CO1. Understand (Level-2) the basic structure, operation and characteristics of digital computer. CO2. Design (Level-6) simple combinational digital circuits based on given parameters. CO3. Understand (Level-2) the working of arithmetic and logic unit as well as the concept of pipelining. CO4. Summarize (Level-2) the hierarchical memory system including cache memories and virtual memory. CO5 Identify (Level-2) the contributions of Indians in the field of computer architecture and related technologies. Note: Level of Bloom's Taxonomy is mentioned in the brackets.
6.	Credit Value	Theory 3 Credits
7.	Total Marks	Max. Marks : 30+70 =100 I Min. Passing Marks: 35

PART B: Content of the Course

No. of Lectures (in hours per week): 3 Hrs. per week

Total No. of Lectures: 45 Hrs.

Module	Topics	No. of
I	Vedic Mathematics in computing: Vedic methods like Nikhilam Sutra and Urdhva- Tiryagbhyam, Fundamentals of Digital Electronics: Number System-Binary, Decimal, Octal, Hexa-Decimal, Conversions, Binary Arithmetic-Addition, Subtraction, Multiplication, Division, Underflow, Overflow, Sign Magnitude, Complements-1's and 2's, Fixed-Point Representation, Floating-Point Representation.	10
II	Boolean Algebra, Reducing Boolean Expression, Logic Gates-AND, OR, NOT, Universal Gates-NAND, NOR, Analog and Digital Signals, Clock Waveform Timing, Map Simplification, K-Map- Two, Three and Four variables.	10
III	Combinational Circuits- Adder, Subtractor, Multiplexer, De-multiplexer, Decoders, Encoders. Binary Codes – Gray Codes, ASCII code, BCD code, EBCDIC, Error Detection Code and Correction Code, Hamming Code.	10

I V	Sequential Circuits - Flip - Flops, SR, D, T, JK, Master-Slave, Registers, Shift Registers- SISO, SIPO, PISO, PIPO, Counters, Instruction, Instruction Format, Instruction Codes, instructions Cycles, Addressing Modes.	10
V	Handshaking, Concepts of RISC, CISC, DMA Data Transfer, Auxiliary Memory, Cache Memory, Associative Memory, Virtual Memory, Indian Contribution.	10
Keywords/Tags: Digital Electronics, Logic Gates, Circuits, Instruction formats, Memory hierarchy.		

Assessment and Evalution		
Suggested Continuous Evalution Methods: Maximum Marks: 100 Continuous Comprehensive Evaluation (CCE): 30 Marks University Exam (UE): 70 Marks		
Internal Assessment: Continuous Comprehensive Evaluation (CCE)	Class Test Assignment/Presentation	Total Marks: 30
External Assessment: University Exam (UE) Time: 03.00 Hours	Section (A): Objective Type Section (B): Short Questions Section (C): Long Questions	Total Marks: 70

CO		**PO**	**PSO**	**Cognitive Level**
CO1	Understand the basic structure, operation and characteristics of digital computer.	PO3	PSO-1	Understanding and Remembering
CO2	Be able to design simple combinational digital circuits based on given parameters.	PO4	PSO-3	Applying and Analysing
CO3	Familiarity with working of arithmetic and logic unit as well as the concept of pipelining.	PO5	PSO-5	Creating, Analysing and Evaluating
CO4	Know about hierarchical memory system including cache memories and virtual memory.	PO4	PSO-3	Creating, Analysing and Evaluating
CO5	Understand concept and advantages of parallel ism, threading, multi-processors and multi-core processors.	PO5	PSO- 5	Understanding and Analysing

PART C: Content of the Course		
No. of Lab. Practical s (in hours per week): 2 Hrs. per week		
Total No. of Labs:		
	Suggestive list of Practical	No. of Labs.
	1. To study basic gates (AND, OR, NOT) and verify their truth tables. 2. To study and verify NAND as Universal gate using IC 7400. 3. To realize basic gate AND from Universal gate NAND. 4. To realize basic gate OR from Universal gate NAND. 5. To realize basic gate N OT from Universal gate NAND. 6. To study and verify NOR as Universal gate 7. To realize basic gate AND from Universal gate NOR. 8. To realize basic gate OR from Universal gate NOR. 9. To realize basic gate NOT from Universal gate NOR. 10. To study Half Adder using basic gates and verify its truth table. 11. To study Full Adder using basic gates and verify its truth table. 12. To verify De-Morgan's First Law Theorem. 13. To verify De-Morgan's Second Law Theorem. 14. To design and construct RS flip Flop using gates and verifies the truth table. 15. To design and construct JK Flip Flop using gates and verifies the truth table.	
	Keywords/Tags: Digital Electronics, Logic Gates, AN D, OR, NOT, IC7486, IC 7400, NAND, NOR, IC 7483, Circuits, Flip Flop, De-Morgan's Theorem.	

PART A: Introduction				
Program: **Under Graduate**		Class: **B.Sc.**	Year: **First Year**	Session: **2025-26**
Subject: **Computer Science**				
1.	Course Code			
2.	Course Title	C-2(TH): **Programming Methodologies & Data Structures (Using C/C++)**		
3.	Course Type (Core Course/Elective/Generic Elective/ Vocational	**Core Course**		
4.	Pre-Requisite (if any)	To study this course, Mathematics of 12th standard is desirable.		
5.	Course Learning Outcomes (CLO)	On completion of this course, learners will be able to: 1. Develop *(Level-6)* simple algorithms and flow charts to solve a problem with programming using top-down design principles; 2. Writing *(Level-6)* efficient and well-structured computer algorithms/programs; 3. Formulate *(Level-6)* iterative solutions and array processing algorithms for problems; 4. Use *(Level-3)* recursive techniques, pointers and searching methods in programming; 5. Implement *(Level-3)* fundamental data structures & accustomed to the description of algorithms in both functional and procedural styles; 6. Understand *(Level-2) the* complexity of basic operations like insert, delete, search on these data structures. 7. Select appropriate *(Level-5)* data structure ~~to~~ suitable to different models; 8. Design *(Level-6)* programs using various data structures including hash tables, Binary and general search trees, heaps, graphs etc.; 9. Assess *(Level-5)* efficiency tradeoffs among different data structure implementations; 10. Implement *(Level-3)* and know the applications of algorithms for searching and sorting etc.; 11. Identify *(Level-2)* the contributions of Indians in the field of programming and data structures. ***Note:*** *Level of Bloom's Taxonomy is mentioned in the brackets.*		
6.	Credit Value	Theory - **4 Credits**		
7.	Total Marks	Max. Marks: **30+70**	Min. Passing Marks: **35**	

PART B: **Content of the Course**		
No. of Lectures (in hours per week): **04 Hrs. per week**		
Total No. of Lectures: **60** Hrs.		
Module	**Topics**	**No. of Lectures**
I	*Relevant Indian Knowledge System (IKS) Inclusions:* Algorithmic thinking in Ancient India. The Panini Grammar System (Ashtadhyayi). The Chandas Shastra (Sanskrit Prosody) a recursive structure, The Brahmagupta Algorithm (7th century CE) an introduction to zero and place value notation. *Data Structures & Computational Methods in Ancient India:* Vedic method of data structuring – preservation of ancient manuscripts (e.g., Rig Veda) using hierarchical structures. Resemblance of Graph Theory with the Indian Temple Architecture (the connectivity principles of temple design and city planning). Resemblance of efficient Sorting & Searching techniques with Ancient Indian classification methods in Ayurveda & Sanskrit texts. The Buddhist Numerical Sorting Method (Bhāskara II). *Introduction to Programming:* Program Concept, Characteristics of Programming, Stages in Program Development, Algorithms, Notations, Design, Flowcharts, Types of Programming Methodologies. *Introduction to C/C++ Programming:* Basic Program Structure in *C/C++*, Data Types, Variables, Constants, Operators and Basic I/O. *Variables:* Declaring, Defining and Initializing Variables, Scope of Variables, Using Named Constants, Keywords, Casting of Data Types, Operators (Arithmetic, Logical and Bitwise), Using Comments in programs, Character I/O (*getc, getchar, putc, putchar* etc.), Formatted and Console I/O [*printf(), scanf(), cin(), cout()*], Using Basic Header Files (*stdio.h, iostream.h, conio.h* etc.), Simple Expressions in *C/C++* (including Unary Operator Expressions, Binary Operator Expressions), Understanding Operators Precedence in Expressions. ***Suggested activities for experiential learning:*** 1. *Comparative Analysis: Research how Panini's grammar rules resemble formal grammar in programming languages.* 2. *Algorithm Simulation: Implement Brahmagupta's place-value system using C/C++.* 3. *Keyword Identification Exercise: Analyze the similarity between Sanskrit syntax and C/C++ keywords (e.g., structure in Sanskrit grammar vs. C/C++ struct).* 4. *Basic I/O Project: Implement a console-based quiz using formatted I/O.*	10

II	Conditional Statements *if* construct, *switch-case* construct. *Iterative Statements: while, do-while,* and *for* loops, use of *break* and *continue* in loops, Using Nested Statements (Conditional as well as Iterative). *Functions:* Top-Down Design, Pre-defined Functions, Programmer defined Functions. Local Variables and Global variables, Functions with Default Arguments, Call-By-Value and Call-By-Reference, Parameters, Recursion. *Introduction to Arrays:* Declaration and Referring Arrays, Arrays in Memory, Initializing Arrays. Arrays in Functions, Multi-Dimensional Arrays. ***Suggested activities for experiential learning:*** 1. *Code Debugging Challenge: Assign buggy C/C++ programs for students to debug and improve.* 2. *Concept Visualization: Use flowcharts and pseudocode tools to map variable types and memory usage.* 3. *Conditional Logic Game: Design a decision-based game using **if-else** and **switch-case**.* 4. *Function Optimization Task: Analyze and optimize recursive vs. iterative function performance.* 5. *Nested Loops Visualization: Represent nested loops using Pascal's Triangle visualization.* 6. *Real-World Decision-Making Simulation: Create a banking/ATM system that demonstrates nested loops and conditional logic.*	08
III	*Structures:* Member Accessing, Pointers to Structures, Structures and Functions, Arrays of Structures. *Unions:* Declaration and Initialization. *Strings:* Reading and Writing Strings, Arrays of Strings, String and Function, Strings and Structure, Standard String Library Functions. *Searching Algorithms:* Linear Search, Binary Search. *File Handling:* Use of files for data input and output, merging and copying files. Ayurvedic Classification System: Map hierarchical classification of medicinal plants to data structures like arrays and pointers, Shulba Sutras for Spatial Computations: Understanding recursive patterns in Shulba Sutras and their application in functions. ***Suggested activities for experiential learning:*** 1. *Mapping Ayurvedic Taxonomy to Data Structures: Represent Ayurvedic classification of herbs using arrays and nested data structures.* 2. *Function Optimization Project: Implement recursive and iterative functions to compare execution time.* 3. *Group seminar and Online quiz based on searching algorithm and file handling*	08

	4. *Vedic Sorting Implementation: Develop a sorting algorithm inspired by Ayurvedic classification techniques.*	
IV	*Data Structure:* Basic concepts, Linear and Non-Linear data structures. *Algorithm Specifications:* Introduction, Recursive algorithms, Data Abstraction, Performance analysis. *Linked List:* Singly Linked Lists, Operations, Concatenating, circularly linked lists-Operations for Circularly linked lists, Doubly Linked Lists- Operations. *Array:* Representation of single, two-dimensional arrays, sparse matrices-array and linked representations. *Stack:* Operations, Array and Linked Implementations, Applications Infix to Postfix Conversion, Postfix Expression Evaluation, Recursion Implementation. *Linked List Concept in Indian Knowledge:* Ancient Sanskrit texts used linked hierarchical structuring for preserving information (e.g., Vedic oral tradition). *Stack Analogy in Nyaya Logic:* Indian logical frameworks used last-in-first-out (LIFO) reasoning similar to stack operations. ***Suggested activities for experiential learning:*** 1. *Chart Preparation: Prepare charts Linked List, Array & Stack* 2. *Stack Simulation Exercise via Role-Play: Implement a LIFO-based task scheduler.* 3. *Linked List Research Assignment: Compare linked list pointer-based structure with ancient manuscript referencing.* 4. *Students will map historical Guru-Shishya Parampara in the form of a singly linked list (E.g., Vyasa → Shuka → Gaudapada → Govindapada → Adi Shankaracharya)* *Implement this as a linked list in C/C++, where each node represents a teacher and links to their disciples.*	12
V	*Queue:* Definition, Operations, Circular Queue-Insertion and Deletion Operations, Dequeue (Double Ended Queue), Priority Queue-Implementation. *Trees:* Representation of Trees, Binary tree, Properties of Binary Trees, Binary Tree Representations- Array and Linked Representations, Binary Tree Traversals, Threaded Binary Trees. *Heap:* Definition, Insertion, Deletion. *Buddhist Numerical Sorting:* Bhaskara II's early classification techniques, *Efficient Searching in Ayurveda:* Ayurvedic medicinal classification principles resemble hashing and tree-based sorting. ***Suggested activities for experiential learning:*** 1. *Queue-based Ticketing System: Develop a queue system (FIFO) for handling real-world ticket processing.*	10

	2. *Search Algorithm Hackathon:* Implement linear, binary, and hashing techniques to solve real-world problems. 3. *Sorting Race:* Students compete to optimize sorting algorithms based on Ayurvedic classification techniques. 4. *Data Organization Challenge:* Create efficient storage models for Ayurveda medicinal records using tree-based structures. 5. *Comparative Study:* Research how Vedic knowledge management compares with modern database indexing.	
VI	*Graphs:* Graph ADT, Graph Representations, Graph Traversals, Searching. *Hashing:* Introduction, Hash tables, Hash functions, Overflow Handling. *Sorting Methods:* Comparison of Sorting Methods, Search Trees - Binary Search Trees, AVL Trees- Definition and Examples. *Tree Representation in Ancient India:* Genealogy (Gotra System) as an early example of hierarchical tree structures. *Graph Connectivity in Temple Design:* Principles of temple planning akin to graph traversal. ***Suggested activities for experiential learning:*** 1. *Family Tree Implementation:* Use binary trees to model ancient Indian lineage systems. 2. *Graph Problem Solving:* Model Indian temple network connectivity using graph algorithms. 3. *Heap Data Structure Exploration:* Implement heap sorting for priority-based Ayurveda classification. 4. *Shortest Path Challenge:* Use Dijkstra's Algorithm to optimize ancient pilgrimage route planning. 5. *Graph Theory Workshop: Study the resemblance of temple architecture to graph connectivity and model it using Graphviz/NetworkX.* 6. *Tree Traversal Experiment: Implement tree traversal to simulate genealogy in Vedic lineage texts.*	10
VII	Indian Contribution to the field: Innovations in India, origin of Julia Programming Language, Indian Engineers who designed new programming languages, open-source languages, Dr. Sartaj Sahni — Computer Scientist - Pioneer of data structures, Murthy's Early Work in Software Development. Julia Programming Language's Indian Origins. ***Suggested activities for experiential learning:*** 1. *Research Presentation:* Students present on Indian-origin computer scientists. 2. *Coding Tribute:* Implement an algorithm inspired by Sartaj Sahni's data structure optimizations. 3. *Innovation Showcase:* Identify Indian-origin open-source projects and contribute to them. 4. *Documentary Screening & Discussion:* Screen a documentary on India's tech evolution and discuss its impact.	2

PART C: Learning Resources

Textbooks, Reference Books, Other Resources

Suggested Readings:

- Lipschutz: Schaum's outline series Data structures, Tata McGraw-Hill
- Problem Solving and Program Design in C, J. R. Hanly and E. B. Koffman, Pearson, 2015
- E. Balguruswamy, "C++ " TMH Publication ISBN 0-07-462038-X
- Herbertz Shield, "C++ The Complete Reference "TMH Publication ISBN 0-07-463880-7
- R. Lafore, "Object Oriented Programming C++"
- N. Dale and C. Weems, Programming and problem solving with C++: brief edition, Jones & Bartlett Learning.
- Adam Drozdek, "Data Structures and algorithm in C++", Third Edition, Cengage Learning.
- Tony Gaddis, Starting Out With Python
- Kenneth A. Lambert, Fundamentals of Python
- Kiparsky, P. (2009). Panini as a Formalist.
- Joseph, G. G. (1991). The Crest of the Peacock: Non-European Roots of Mathematics.
- Bhaskara II (12th Century), Lilavati and Bijaganita.
- Staal, F. (2006). The Science of Language and Logic in India.
- Pingree, D. (1978). Mathematical Astronomy in India.
- Kosambi, D. D. (1948). The Culture and Civilization of Ancient India.
- Sahni, S. (2005). Data Structures, Algorithms, and Applications in C++.

Suggested online resources:

http://www.ndl.gov.in/he_document/ekumbh/97
http://www.ndl.gov.in/he_document/nptel/IN_N_1_C_S_a_E_9093_N_P_D_S_a_A_u_P_12265_12266

https://archive.nptel.ac.in/courses/106/105/106105171/
https://archive.nptel.ac.in/courses/106/105/106105234/
https://archive.nptel.ac.in/courses/106/101/106101208/
https://archive.nptel.ac.in/courses/106/106/106106133/
https://epgp.inflibnet.ac.in/Home/ViewSubject?catid=fBYckQKJvP3a/8Vd3L08tQ==
https://nptel.ac.in/courses/106105151

Part D: Assessment and Evaluation

Suggested Continuous Evaluation Methods:
Maximum Marks: **100**
Continuous Comprehensive Evaluation (CCE): **30** Marks
University Exam (UE): **70** Marks

Internal Assessment: Continuous Comprehensive Evaluation (CCE)	Class Test Assignment/Presentation	**Total Marks: 30**
External Assessment: University Exam (UE) Time: 03.00 Hours	Section (A): Objective type Section (B): Short Questions Section (C): Long Questions	**Total Marks: 70**

PART A: Introduction				
Program: **Under Graduate**		Class: **B.Sc.**	Year: **First Year**	Session: **2025-26**
Subject: **Computer Science**				
1.	Course Code			
2.	Course Title	C-2(PR): **Programming Methodologies & Data Structures (Using C/C++) (Lab)**		
3.	Course Type (Core Course/Elective/Generic Elective/ Vocational	Core Course		
4.	Pre-Requisite (if any)	To study this course, Mathematics of 12thstandard is desirable.		
5.	Course Learning Outcomes (CLO)	On completion of this course, learners will be able to: 1. Develop *(Level-6)* simple algorithms and flow charts to solve a problem with programming using top-down design principles. 2. Writing *(Level-6)* efficient and well-structured computer algorithms/programs. 3. Formulate *(Level-6)* iterative solutions and array processing algorithms for problems. 4. Apply *(Level-4)* recursive techniques, pointers and searching methods in programming. 5. Select *(Level-5)* appropriate data structure suitable to different models; 6. Implement *(Level-3)* and know the applications of algorithms for searching and sorting etc.; ***Note:*** *Level of Bloom's Taxonomy is mentioned in the brackets.*		
6.	Credit Value	Practical **-2 Credits**		
7.	Total Marks	Max. Marks: **100**	Min. Passing Marks: **35**	

PART B: **Content of the Course**	
No. of Lab Practical (in hours per week): **2 per week**	
Total No. of Lab.: **60 Hrs.**	
Suggestive list of Practical	No. of Labs.

Given the problem statement, students are required to formulate problem, develop flowchart/algorithm, write code in **C/C++**, execute and test it. Students should be given assignments on following:

1. a. To learn elementary techniques involving arithmetic operators and mathematical expressions, appropriate use of selection (if, switch, conditional operators) and control structures.

   b. Learn how to use functions and parameter passing in functions, writing recursive programs.
2. Write a program to swap the contents of two variables.
3. Write a program for finding the roots of a Quadratic Equation.
4. Write a program to find area of a circle, rectangle, square using switch case.
5. Write a program to check whether a given number is even or odd.
6. Write a program to print table of any number.
7. Write a program to print Fibonacci series.
8. Write a program to find factorial of a given number.
9. Write a program to convert decimal (integer) number into equivalent binary number.
10. Write a program to check given string is palindrome or not.
11. Write a program to perform multiplications of two matrices.
12. Write a program to print digits of entered number in reverse order.
13. Write a program to print sum of two matrices.
14. Write a program to print multiplication of two matrices.
15. Write a program to generate even/odd series from 1 to 100.
16. Write a program whether a given number is prime or not.
17. Write a program for call by value and call by reference.
18. Write a program to generate a series 1+1/1 !+2/2!+3/3!+----------+n/n!
19. Write a program to create a pyramid structure

    *
    **
    ***
    ****
20. Write a program to create a pyramid structure.

    1
    12
    123
    1234
21. Write a program to check entered number is Armstrong or not.
22. Write a program for traversing an Array.
23. Write a program to input N numbers, add them and find average.
24. Write a program to find largest element from an array.
25. Write a program for Linear search.
26. Write a program for Binary search.
27. Write a program for Bubble sort.
28. Write a program for Selection sort.

PART C: Learning Resources

Textbooks, Reference Books, Other Resources

Suggested Readings:

- Problem Solving and Program Design in C, J. R. Hanly and E. B. Koffman, Pearson, 2015
- E Balguruswamy, "C++ " TMH Publication ISBN 0-07-462038-X
- Herbertz Shield, "C++ The Complete Reference "TMH Publication ISBN 0-07-463880-7 R. Lafore, "Object Oriented Programming C++"
- N. Dale and C. Weems, "Programming and problem solving with C++", brief edition, Jones & Bartlett Learning.
- Adam Drozdek, "Data Structure and Algorithm in C++", Third Edition, Cengage Learning.
- Sartaj Sahani, "Data Structure, Algorithm and Applications with C++", McGraw Hill.
- Robert L. Kruse, Data Structure and Program Design in C++" Pearson.
- D.S. Malik, Data Structure using C++, Second Edition, Cengage Learning.
- M. A. Weiss, Data structures and Algorithm Analysis in C, 2nd edition, Pearson.
- Lipschutz: Schaum's outline series Data structures, Tata McGraw-Hill

Suggestive digital platform web links:

https://www.yout-ube.com/watch?v=BClS40yzssA
https://www.yout-ube.com/watch?v=vLnPwxZdW4Y&vl=en
https ://www.yout-ube.com/watch?v=Umm I ZQ5 ItZw
https://nptel.ac.in/courses/106/ 106/ 106106127/

Suggested equivalent online courses:

https://nptel.ac.in/courses/106/105/ 106105 1 5 1 /
https://nptel.ac.in/courses/106/105/ 106105 1 7 1/

PART D: Assessment and Evaluation

Suggested Continuous Evaluation Methods:

Internal Assessment	**Marks**	**External Assessment**	**Marks**
Class Interaction/Quiz	**NIL**	Viva Voce on Practical (20 marks)	**100**
Attendance		Practical Record File (20 marks)	
Assignments (Charts/Model/Seminars / Technology Dissemination/ Excursion/ Lab visit/ Industrial Visit)		Table Work / Exercise Assigned (60 marks)	
	Total Marks: 100		

Prof. Navita Shrivastava
Chairman Board of Studies

PART A: **Introduction**			
Program: **Under Graduate**	Class: **B.Sc.**	Year: **First Year**	Session: **2025-26**
Subject: **Computer Science**			

1.	Course Code	
2.	Course Title	C-3(TH): **Operating System & Office Tools**
3.	Course Type (Core Course/Elective/Generic Elective/ Vocational	**Core Course**
4.	Pre-Requisite (if any)	To study this course, Mathematics of 12th standard is desirable.
5.	Course Learning Outcomes (CLO)	On completion of this course, learners will be able to: 1. Understand *(Level-2)* the fundamental concepts of operating systems, their functions, and architectures; 2. Analyze *(Level-4)* process management, scheduling, concurrency, and memory management techniques; 3. Understand *(Level-2)* the file systems, I/O management, security, and virtualization; 4. Develop *(Level-6)* proficiency in state-of-the-art office productivity tools, including word processing, spreadsheets, presentations, database management, and collaboration tools; 5. Explain *(Level-2)* India's contributions towards computing, knowledge structuring, and resource management techniques. ***Note:*** *Level of Bloom's Taxonomy is mentioned in the brackets.*
6.	Credit Value	Theory -**4 Credits**
7.	Total Marks	Max. Marks: **30+70** Min. Passing Marks: **35**

PART B: Content of the Course

No. of Lectures (in hours per week): **04 Hrs. per week**

Total No. of Lectures: **60 Hrs.**

Module	Topics	No. of Lectures
I	*Introduction to Operating Systems:* Definition, Functions, Evolution, Types (Batch, Multiprogramming, Time-Sharing, Distributed, Real-Time, Embedded). *Resource allocation concepts in Indian Philosophy:* Pancha Kosha theory - layered abstraction akin to OS design, Manuscript storage in ancient Indian libraries (Takshashila & Nalanda) as an analogy for memory management. ***Suggested activities for experiential learning:*** 1. Compare modern OS resource allocation techniques with the layered structure of Pancha Kosha. 2. *Group Discussion & Concept Mapping*: Explore how ancient manuscript storage systems parallel modern memory management. 3. *Simulation Exercise:* Develop a simulation that models resource allocation algorithms (e.g., CPU scheduling) in an OS environment. 4. *Conceptual Flowchart Development:* Create flowcharts that map the evolution of OS design, linking abstract layers to practical resource management.	8

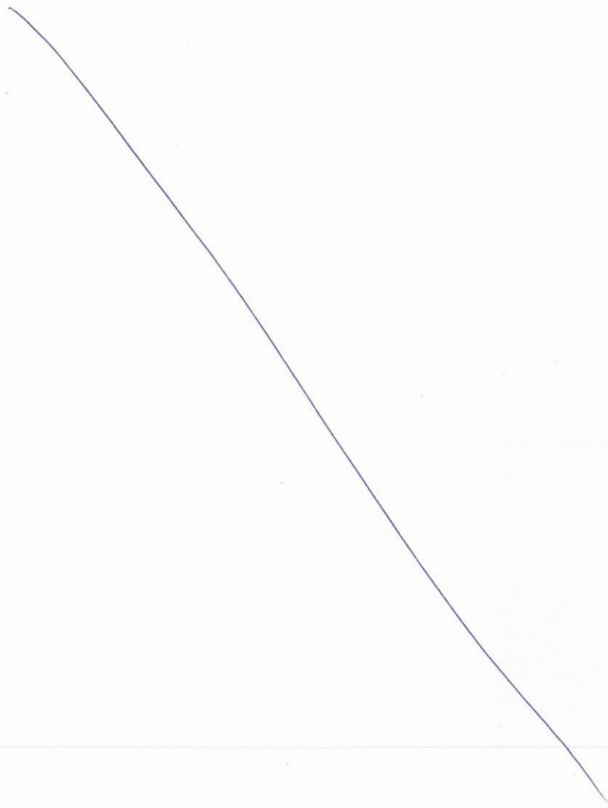
II	*Process Management:* Process Concept, Process Scheduling (FCFS, SJF, Round Robin, Priority Scheduling), Threads, Inter-process Communication (IPC). *Task scheduling in Vedic rituals:* Yagna sequences similar to scheduling algorithms, Efficient computation models in Indian astronomy (Aryabhata's planetary motion calculations). ***Suggested activities for experiential learning:*** 1. *Simulation Modeling:* Develop a simulation of process scheduling (e.g., FCFS, SJF) and compare its performance with the sequential order observed in Vedic rituals. 2. *Real-life Analogy:* Ask students to map everyday situations (e.g., waiting in a queue, dividing tasks among teammates) to process scheduling concepts. 3. *Role-Playing Exercise (Core CS):* Organize a role-play where students act out process scheduling to understand priorities and time-sharing. 4. *Thread Race Game: Create a hands-on game where students simulate threads competing for CPU time. Use physical tokens or cards to represent tasks and resources.*	8
III	*Concurrency & Synchronization:* Process Synchronization, Critical Sections, Deadlocks (Prevention, Avoidance, Detection), Semaphores, CPU Scheduling. Parallel execution concepts in Indian astronomical texts (e.g., Surya Siddhanta) compared to modern concurrent processing. ***Suggested activities for experiential learning:*** 1. *Interactive Simulation:* Build a simulation to model concurrent process execution and explore synchronization techniques (e.g., semaphores, monitors). 2. *Hands-On Lab:* Design and test synchronization protocols to resolve common issues like race conditions and deadlocks. 3. *Role-Playing Activity:* Have students simulate process synchronization, with roles assigned as processes and synchronization tools. 4. Organize a debate in which students argue the concept of "parallelism" in the ***Surya Siddhanta*** versus modern computational models. 5. *Group Discussion:* Discuss how tasks can be divided and executed simultaneously to reduce overall execution time.	8
IV	*Memory Management:* Memory Hierarchy, Virtual Memory, Paging, Segmentation, Fragmentation, Thrashing. Indexing & storage techniques in ancient Indian texts (Rigveda's indexing method similar to hierarchical memory management). ***Suggested activities for experiential learning:*** 1. *Simulation Project:* Develop a simulator for virtual memory management and paging, drawing analogies with ancient hierarchical indexing. 2. *Comparative Research Assignment:* Investigate and present on the similarities between ancient manuscript organization and modern cache memory systems. 3. *Visualization Workshop:* Design flowcharts or diagrams that illustrate memory allocation and fragmentation concepts. 4. *Hands-On Implementation:* Implement a memory management algorithm to study fragmentation and allocation strategies in a simulated OS environment.	8

V	*File Systems & I/O Management:* File Organization, Allocation Methods (Contiguous, Linked, Indexed), Directory Structure, Disk Scheduling (FCFS, SSTF, SCAN, LOOK). Knowledge documentation in Indian scriptures (Palm-leaf manuscripts stored using structured indexing). ***Suggested activities for experiential learning:*** 1. *File System Design Project*: Design file system architecture inspired by the organized storage of ancient manuscripts. 2. *Lab Exercise:* Simulate different file allocation methods (contiguous, linked, indexed) using available tools. 3. *Group Project:* Develop a directory management system incorporating modern OS file system concepts. 4. *Case Study Discussion:* Analyze real-world file system challenges and propose solutions, integrating both historical insights and modern techniques.	8
VI	*Office Productivity Tools:* Word Processing, Spreadsheets, Presentations, Database Management, Email & Collaboration Tools. Emphasizing latest versions of widely used office productivity suites such as MS Office 365 (with AI support), Google Workspace, LibreOffice, and any emerging tools. The focus will be on adaptability to evolving digital technologies. ***Suggested activities for experiential learning:*** 1. *Tool Comparison Workshop:* Compare different office productivity suites (e.g., document editors, spreadsheets, presentations) based on functionality and usability. 2. *Integrated Project:* Develop a comprehensive project that requires the creation and management of documents, spreadsheets, and presentations to simulate real-world business scenarios. 3. *Collaborative Simulation:* Organize a virtual group exercise using modern collaboration tools to solve a practical problem. 4. *Expert Session:* Invite an industry professional to demonstrate advanced features and discuss emerging trends in digital productivity.	8
VII	*Security & Virtualization:* Basics of Computer Security, User Authentication, Access Control, Cloud Computing, Virtual Machines, Containers. Ancient Indian encryption techniques (Kautilya's Arthashastra on cryptography), Role of security in knowledge preservation (e.g., coded messages in ancient war strategies). ***Suggested activities for experiential learning:*** 1. *Encryption Lab:* Implement basic encryption and decryption algorithms inspired by ancient techniques and compare with modern methods. 2. *Virtualization Simulation:* Develop a simulation to understand virtualization concepts and the isolation of virtual environments. 3. *Security Policy Workshop:* Draft a security policy for a hypothetical organization, integrating both modern authentication protocols and historical secure communication practices. 4. *Case Study Analysis:* Examine historical instances of secure communication and relate them to contemporary security challenges.	8

VIII	*Indian Contributions to Operating Systems & Computing:* Achievements of Raj Reddy (AI & Speech Recognition), Pravin Bhagwat (Wi-Fi Protocols), Satyam Priyadarshy (Cloud Computing), and India's Role in Open Source. India's innovations in software development, networking, and artificial intelligence. ***Suggested activities for experiential learning:*** 1. *Research Presentation:* Prepare detailed presentations on the achievements of key Indian computing pioneers and their impact on modern OS development. 2. *Innovation Workshop:* Organize a mini-hackathon where students develop small projects inspired by the innovations of these pioneers. 3. *Panel Discussion:* Host a panel with industry experts and academicians to discuss the influence of Indian contributions on global computing trends. 4. *Documentary Screening & Discussion:* Watch a documentary on India's technological advancements, followed by a reflective group discussion on lessons learned and future prospects.	4

PART C: **Learning Resources**
Textbooks, Reference Books, Other Resources
Suggested Readings: • Silberschatz, Galvin, Gagne: Operating System Concepts, Wiley. • William Stallings: Operating Systems: Internals & Design Principles, Pearson. • Andrew S. Tanenbaum: Modern Operating Systems, Pearson. • Rajaraman: Fundamentals of Computers, PHI Learning. • ITL Education: Introduction to Information Technology, Pearson. • S. (2004). Ancient Indian Knowledge Systems and Their Relevance to Modern Technology. • Ghosh, A. (2001). Indian Philosophy and Organizational Systems. • Nair, A. (2015). Indian Innovators in Computing. • Rao, S. (2013). Technological Contributions from India. • Mehta, P. (2006). Cryptography in Arthashastra. • Desai, V. (2012). Ancient Security Practices and Modern Cryptography. • Banerjee, S. (2002). Documentation Systems in Ancient India. • Singh, R. (2008). File Systems: An Indian Historical Perspective. • Kumar, D. (2007). Ancient Indexing Methods and Modern Memory Systems. • Jain, M. (2003). Memory and Manuscripts: An Indian Perspective. • Reddy, N. (2010). Parallelism in Ancient Indian Astronomy. • Menon, K. (2008). Concurrency Concepts in Historical Context. • Gupta, P. (1999). Aryabhata's Contributions to Astronomy and Computing. • Sharma, R. (2005). Ancient Scheduling Techniques in Indian Rituals. ***Suggested online Learning Resources:*** ○ NPTEL Course: Operating System Principles – NPTEL Link ○ Office Tools (Latest MS Office 365, AI Support) ○ https://nptel.ac.in/courses/106106144 ○ https://archive.nptel.ac.in/courses/106/105/106105214/ ○ https://epgp.inflibnet.ac.in/Home/ViewSubject?catid=fBYckQKJvP3a/8Vd3L08tQ==

Part D: Assessment and Evaluation		
Suggested Continuous Evaluation Methods: Maximum Marks: **100** Continuous Comprehensive Evaluation (CCE): **30** Marks University Exam (UE): **70** Marks		
Internal Assessment: Continuous Comprehensive Evaluation (CCE)	Class Test Assignment/Presentation	**Total Marks: 30**
External Assessment: University Exam (UE) Time: 03.00 Hours	Section (A): Objective type Section (B): Short Questions Section (C): Long Questions	**Total Marks: 70**



PTO

PART A: **Introduction**			
Program: **Under Graduate**	Class: **B.Sc.**	Year: **First Year**	Session: **2025-26**
Subject: **Computer Science**			

1.	Course Code		
2.	Course Title	C-3(PR): **Operating System & Office Tools (Lab)**	
3.	Course Type (Core Course/Elective/Generic Elective/Vocational)	**Core Course**	
4.	Pre-Requisite (if any)	To study this course, Mathematics of 12thstandard is desirable.	
5.	Course Learning Outcomes(CLO)	On completion of this course, learners will be able to: 1. Perform *(Level-6)* the simulation for process scheduling, disk-scheduling, memory management techniques. 2. Perform *(Level-6)* the dead-lock detection using modeling. 3. Perform *(Level-6)* the file-handling using Linux commands and shell scripting. 4. Perform *(Level-6)* the formatting of text on various word processing tools. 5. Create *(Level-6)* spreadsheet using formula and pivot tables. 6. Create *(Level-6)* and manipulate *(Level-3)* the data using database tools. 7. Create *(Level-6)* presentations using AI tools in quick and smart way. 8. Manage *(Level-6)* online-meetings through Google meet, MS-teams, Zoom etc. applications. ***Note:*** *Level of Bloom's Taxonomy is mentioned in the brackets*	
6.	Credit Value	Practical - **2 Credits**	
7.	Total Marks	Max. Marks: **100**	Min. Passing Marks: **35**

PART B: **Content of the Course**		
No. of Lab. Practical (in hours per week): **2 Hrs. per week**		
Total No. of Labs: **60 Hrs.**		
	Suggestive list of Practical	No. of Labs.
	I - Office Tools: 1 *Word Processing Tools:* Document formatting, referencing, macros (Using latest versions of MS Word, Google Docs, LibreOffice). • Create a document and apply different Editing options. • Create Banner for your college. • Design a Greeting Card using Word Art for different festivals. • Design your Bio data and use page borders and shading. • Create a document and insert header and footer, page title, date, time, apply various page formatting features etc. • Implement Mail Merge. • Insert a table into a document and try different formatting options for the table. 2 *Spreadsheet Tools:* Advanced formulas, data visualization, pivot tables (Using latest Excel, Google Sheets). • Design your class Time Table. • Prepare a Mark Sheet of your class result. • Prepare a Salary Slip of an employee of an organization. • Prepare a bar chart & pie chart for analysis of Election Results. • Prepare a generic Bill of a Super Market. • Work on the following exercises on a Workbook: • Copy an existing Sheet • Rename the old Sheet • Insert a new Sheet into an existing Workbook • Delete the renamed Sheet. • Prepare an Attendance sheet of 10 students for any 6 subjects of your syllabus. Calculate their total attendance, total percentage of attendance of each student & average of attendance. • Create a worksheet of Students list of any 4 faculties and perform following database functions on it. ○ Sort data by Name ○ Filter data by Class ○ Subtotal of no. of students by Class. 3 *Presentation Tools:* AI-assisted slide design, animation, transitions (Using PowerPoint, Google Slides). • Design a presentation of your institute using auto content wizard, design template and blank presentation. • Design a presentation illustrating insertion of pictures, Word Art and ClipArt. • Design a presentation, learn how to save it in different formats,copying and opening an existing presentation. • Design a presentation illustrating insertion of movie, animation and sound. • Illustrate use of custom animation and slide transition (using different effects). • Design a presentation using charts and tables of the marks obtained in class.	

Shrivastava

	4 *Database Tools:* Create and manage databases in MS Access or cloud-based tools. 5 *Collaboration Tools:* Hands-on with Microsoft Teams, Slack, Zoom, Google Meet, and emerging AI-powered collaboration tools II -**Operating System:** 1 Hands-on process scheduling simulation (FCFS, Round Robin, Priority Scheduling). 2 Deadlock detection & prevention simulation using system modelling. 3 Memory management simulation – Implement paging & segmentation. 4 Disk scheduling algorithms – SSTF, SCAN, LOOK using programming. 5 Linux Commands & Shell Scripting – File handling, user management.	

PART C: **Learning Resources**
Textbooks, Reference Books, Other Resources
Suggested Readings: 1. Silberschatz, Galvin, Gagne: Operating System Concepts, Wiley. 2. William Stallings: Operating Systems: Internals & Design Principles, Pearson. 3. Andrew S. Tanenbaum: Modern Operating Systems, Pearson. 4. Rajaraman: Fundamentals of Computers, PHI Learning. 5. ITL Education: Introduction to Information Technology, Pearson. *Suggested Digital Learning Resources:* • NPTEL Course: Operating System Principles • YouTube Lectures: *Operating System Fundamentals* *Office Tools (Latest MS Office 365, AI Support)*

PART D: Assessment and Evaluation

Suggested Continuous Evaluation Methods:

Internal Assessment	**Marks**	**External Assessment**	**Marks**
Class Interaction/Quiz	**NIL**	Viva Voce on Practical (20 marks)	**100**
Attendance		Practical Record File (20 marks)	
Assignments (Charts/Model/Seminars / Technology Dissemination/ Excursion/ Lab visit/ Industrial Visit)		Table Work / Exercise Assigned (60 marks)	
	Total Marks: 100		

Prof. Navita Shrivastava
Chairman Board of Studies

ST. ALOYSIUS COLLEGE(AUTONOMOUS), JABALPUR

Reaccredited 'A++ 'Grade by NAAC(CGPA:3.58/4.00) College with Potential for Excellence by UGC

DST-FIST Supported & STAR College Scheme by DBT

PART C: Learning Resources
Textbooks, Reference Books, Other Resources
Suggested Readings: • Lipschutz: Schaum's outline series Data structures, Tata McGraw-Hill • Problem Solving and Program Design in C, J. R. Hanly and E. B. Koffman, Pearson, 2015 • E. Balguruswamy, "C++ " TMH Publication ISBN 0-07-462038-X • Herbertz Shield, "C++ The Complete Reference "TMH Publication ISBN 0-07-463880-7 • R. Lafore, 'Object Oriented Programming C++" • N. Dale and C. Weems, Programming and problem solving with C++: brief edition, Jones & Bartlett Learning. • Staal, F. (2006). The Science of Language and Logic in India. • Joseph, G. G. (1991). The Crest of the Peacock: Non-European Roots of Mathematics. • Bhaskara II (12th Century). Lilavati and Bijaganita. • Pingree, D. (1978). Mathematical Astronomy in India. • Kosambi, D. D. (1948). The Culture and Civilization of Ancient India. • Sahni, S. (2005). Data Structures, Algorithms, and Applications in C++. Saggestive Digital Platform web links: http://www.ndl.gov.in/he_document/ekumbh/97 http://www.ndl.gov.in/he_document/nptel/IN_N_1_C_S_a_E_9093_N_P_D_S_a_A_u_P_12265_12266 https://archive.nptel.ac.in/courses/106/101/106101208/ https://archive.nptel.ac.in/courses/106/106/106106133/ https://epgp.inflibnet.ac.in/Home/ViewSubject?catid=fBYckQKJvP3a/8Vd3L08tQ== https://nptel.ac.in/courses/106105151 https://archive.nptel.ac.in/courses/106/106/106106145/

Part D: Assessment and Evaluation		
Suggested Continuous Evaluation Methods: Maximum Marks: **100** Continuous Comprehensive Evaluation (CCE): **30** Marks University Exam (UE): **70** Marks		
Internal Assessment: Continuous Comprehensive Evaluation (CCE)	Class Test Assignment/Presentation	**Total Marks: 30**
External Assessment: University Exam (UE) Time: 03.00 Hours	Section (A): Objective type Section (B): Short Questions Section (C): Long Questions	**Total Marks: 70**

PART A: **Introduction**				
Program: **Under Graduate**	Class: **B.Sc.**	Year: **First Year**	Session: **2025-26**	
Subject: **Computer Science**				
1.	Course Code			
2.	Course Title	M-2(PR): **Programming Methodology (Lab)**		
3.	Course Type (Core Course/Elective/Generic Elective/ Vocational	**Minor Course**		
4.	Pre-Requisite (if any)	To study this course, Mathematics of 12th standard is desirable.		
5.	Course Learning Outcomes (CLO)	On completion of this course, learners will be able to: 1. Develop *(Level-6)* simple algorithms and flow charts to solve a problem with programming using top-down design principles; 2. Writing *(Level-6)* efficient and well-structured computer algorithms/programs; 3. Formulate *(Level-6)* iterative solutions and array processing algorithms for problems; 4. Use *(Level-3)* recursive techniques, pointers and searching methods in programming. ***Note:*** *Level of Bloom's Taxonomy is mentioned in the brackets*		
6.	Credit Value	Practical - **01 Credit**		
7.	Total Marks	Max. Marks: **100**	Min. Passing Marks: **35**	

PART B: **Content of the Course**

No. of Lab Practical (in hours per week):
Yearly System = **01** Hrs. per week **OR** Semester System = **02** Hrs. per week

Total No. of Lab.: **30 Hrs**

	Suggestive list of Practical	No. of Labs.
	Given the problem statement, students are required to formulate problem, develop flowchart/algorithm, write code in C++, execute and test it. Students should be given assignments on following: 1. To learn elementary techniques involving arithmetic operators and mathematical expressions, appropriate use of selection (if, switch, conditional operators) and control structures 2. Write a program to swap the contents of two variables. 3. Write a program for finding the roots of a Quadratic Equation. 4. Write a program to find area of a circle, rectangle, square using switch case.	

5. Write a program to check whether a given number is even or odd.
6. Write a program to print table of any number.
7. Write a program to print Fibonacci series.
8. Write a program to find factorial of a given number.
9. Write a program to convert decimal (integer) number into equivalent binary number.
10. Write a program to check given string is palindrome or not.
11. Write a program to perform multiplications of two matrices.
12. Write a program to print digits of entered number in reverseorder.
13. Write a program to print sum of two matrices.
14. Write a program to print multiplication of two matrices.
15. Write a program to generate even/odd series from 1 to 100.
16. Write a program whether a given number is prime or not.
17. Write a program for call by value and call by reference.
18. Write a program to generate a series 1+1/1!+2/2!+3/3!+----------+n/n!
19. Write a program to create a pyramid structure

```
*
**
***
****
```

20. Write a program to create a pyramid structure.

```
1
12
123
1234
```

21. Write a program to check entered number is Armstrong or not.
22. Write a program for traversing an Array.
23. Write a program to input N numbers, add them and find average.
24. Write a program to find largest element from an array.

PART C: Learning Resources

Textbooks, Reference Books, Other Resources

Suggested Readings:

- Problem Solving and Program Design in C, J. R. Hanly and E. B. Koffman, Pearson, 2015
- E Balguruswamy, "C++ " TMH Publication ISBN 0-07-462038-X
- Herbertz Shield, "C++ The Complete Reference "TMH Publication ISBN 0-07-463880-7
  R. Lafore, 'Object Oriented Programming C++"
- N. Dale and C. Weems, Programming and problem solving with C++: brief edition, Jones & Bartlett Learning.

Suggestive digital platform web links: https://www.youtube-nocookie.com/embed/BClS40yzssA?playlist=BClS40yzssA&autoplay=1&iv_load_policy=3&loop=1&start= https://www.youtube-nocookie.com/embed/vLnPwxZdW4Y?playlist=vLnPwxZdW4Y&autoplay=1&iv_load_policy=3&loop=1&start= https://nptel.ac.in/courses/%20106106127
Suggested equivalent online courses: https://nptel.ac.in/courses/%20106105%201%205%201 https://nptel.ac.in/courses/%20106105%201%207%201 https://onlinecourses.swayam2.ac.in/

PART D: Assessment and Evaluation

Suggested Continuous Evaluation Methods:

Internal Assessment	Marks	External Assessment	Marks
Class Interaction/Quiz	NIL	Viva Voce on Practical (20 marks)	100
Attendance		Practical Record File (20 marks)	
Assignments (Charts/Model/Seminars / Technology Dissemination/ Excursion/ Lab visit/ Industrial Visit)		Table Work / Exercise Assigned (20 marks)	
	Total Marks: 100		

Prof. Navita Shrivastava
Chairman Board of Studies

Faculty of Science

Bachelor of Science (B.Sc.), III Semester

SUBJECT: COMPUTER SCIENCE

Paper-Major/Minor

Computer Networks & Information Security

Course Outcomes

CO. No.	Course Outcomes	Cognitive Level
CO 1	Understand the uses and types of computer networks, including broadband, mobile, wireless, and enterprise networks.	U, R
CO 2	Learn about PAN, LAN, MAN, and WAN technologies.	U, R, An
CO 3	Gain knowledge of OSI and TCP/IP models, data link layer protocols, and switching techniques.	U, R, An
CO4	Master routing algorithms, IP addressing, and congestion management.	U, An, Ap
CO5	Understand security principles, types of attacks, and security technologies like firewalls and cryptographic tools.	An, AP, R, U

Credit and Marking Scheme

	Credits	Marks		Total Marks
		Internal	External	
Theory	4	40	60	**100**
Practical	2	40	60	**100**
Total	**6**	**200**		

Evaluation Scheme

	Marks	
	Internal	**External**
Theory	3 Internal Exams of 20 Marks (During the Semester) (Best 2 will be taken)	1 External Exams (At the End of the Semester)
Practical	3 Internal Exams (During the Semester) (Best 2 will be taken)	1 External Exams (At the End of the Semester)

Bachelor of Science (B.Sc.) **Subject: Computer Science**

III Semester **Paper: Major/Minor, Computer Networks & Information Security**

Content of the Course

No. of Lectures (in hours per week): 2 Hrs. per week

Total No. of Lectures: 60 Hrs. **Maximum Marks:** 60

Units	Topics	No. of Lecture
I	Introduction to Computer Network: Use of Computer network: Access to information, person-to-person communication electronic commerce, internet of things. Types of computer networks: Broadband access network, Mobile and wireless network, content delivery network, transit network, Enterprise network. Network Technology: Personal Area Network Local Area Network, Metropolitan Area Network, Wide Area Network, example of network (Internet, Mobile network, wireless network); Reference Model: OSI, TCP/IP, Critique of the OSI and TCP/IP reference models.	10
II	Physical Layer: Guided Transmission Media: Twisted pairs, coaxial cable, Fiber Optics; Wireless transmission: The electromagnetic spectrum, frequency hopping spread spectrum, direct sequence, spread spectrum, ultra deb communication; Cellular Network: Common concepts-cells, handoff, 1G 2G,3G,4G & 5G technology.	10
III	Data Link Layer: Service Provided to Network Layer: Data Link Control: Framing, Flow and Error Control; Error detecting codes, Error-correcting codes; Data Link Protocols: Basic transmission and receipt, simplex, full duplex, sliding window protocol, Point-to-Point Protocol. Switching Techniques: Packet Switching, Circuit Switching, Datagram Networks, VirtualCircuit Networks. Network Devices & Drivers: Router, Modem, Repeater, Hub, Switch, Bridge and Gateway (fundamental concepts)	10
IV	Network Layer: Routing Algorithm: Optimality, Principal of Shortest path algorithm, Flooding, Distance Vector Routing, Broadcast Routing; Congestion in network, traffic management approaches; IP addresses, IPv4 Addresses, IP v6 Addresses. Virtual Circuit Networks: Frame relay and ATM, Transport Layer: Process- Process Delivery; UDP, TCP. Application Layers: DNS, SMTP, POP, FTP, http and https. Basics of Wi-Fi (Fundamental concepts Only).	15
V	Network Security and Information Security: Fundamentals of network and information security: principles of security and attack. Security Goals (Confidentiality, Integrity, and Availability). Overview of Security Threats and Vulnerability: Types of attacks on Confidentiality, Integrity and Availability. Vulnerability and Threats: Phishing Attacks, E-mail threats, web threats, Intruders and Hackers, Insider threats, SQL injection Attacks, Ransomware. Malware: Worms, Virus Spams, Adware, Spyware, Trojans. Security Technology: Firewalls, intruding detection and prevention systems, Scanning and Analysis Tools: Biometric access controls, Cipher methods, cryptographic algorithms, cryptographic tools.	15

References

Textbooks:

- Andrew S. J. Wetherall, Computer Networks, 6th Edition, (2021), Pearson.
- J Mattord, Principles of Information Security, Fourth Edition, 6th Indian Reprint.
  - Praveen Kinnar Shur la, Surya Prakash Tripathi, Ritendra Goe 1 "Introduction to Information Security and Cyber Laws", 2014. Dreamtech Press.
- Books published by Hindi Granth Academy, Bhopal.

Reference Books:

- Kurose James F., Ross Keith W., Computer Networking, A 4 op-Down Approach, Sixth Edition, 2017. Pearson
- Micki Krausc. Harold F. Tipton, Handbook of Information Security Management, Vol. 1-3, CRC Press LLC.

- B. A. Forouzan: Data Communications and Networking. Fourth edition, TMH Publishing Company Ltd.
- Basta W. Halton, Computer Security: Concepts, Issues and Implementation, Cengage Learning India.

List of Practical

- Study of UTP cable
  - Color code of UTP cable Categories of UTP n/w cable
  - Shielding of n/w cable
  - Maximum length for which data cable can be usedCrimping of RJ45 connector and punching of datacab1e
- Knowledge of Structured Cabling and its componentsInformation
  Information outlet with box
  - Network Rack (4U, 6U, 9U, 12U, 24U, 32U, 42U)
  - Patch Panel
  - Rack Management
- Study of Optical Fiber cable
  - Different cores of OF C (6 core, 12, 24 core) Multimode & Single mode OFC cable Shielding ofOFC
  - Splicing/Termination of OFC. OTDR Testing
  - LIU fix
  - LIU management (pigtail/fiber patchcord) and MediaConverter.
  - FP module.
  - Rules of OFC laying
- Use of tools
  - Crimping tool
  - Punching tool
  - Nose plier
  - Wire stripping and cable cutter
  - Multi-meter
- Configuration/management of Local Area Network
  - Implementation of File and printer sharing Installationof ft server and client
  - Connect the computers to Local Area Network.
  - Configuring Class, A IP address on LAN Connection inComputer LAB and use the following tools:
    - Ping, ipconfig, getmac, hostname, nslookup,tracert, systeminfo.
    - routing using packet tracer software
    - Dynamic routing using packet tracer
    - Implementation of Subnetting in Class A, B, C
    - Ping between 2 s2'stems using lPv6

Faculty of Science

Bachelor of Science (B.Sc.), III Semester

SUBJECT: COMPUTER SCIENCE

Paper-Elective

Computer Networks & Information Security

Course Outcomes

CO. No.	Course Outcomes	Cognitive Level
CO 1	Understand the uses and types of computer networks, including broadband, mobile, wireless, and enterprise networks.	U, R
CO 2	Learn about PAN, LAN, MAN, and WAN technologies.	U, R, An
CO 3	Gain knowledge of OSI and TCP/IP models, data link layer protocols, and switching techniques.	U, R, An
CO4	Master routing algorithms, IP addressing, and congestion management.	U, An, Ap
CO5	Understand security principles, types of attacks, and security technologies like firewalls and cryptographic tools.	An, AP, R, U

Credit and Marking Scheme

	Credits	Marks		Total Marks
		Internal	External	
Theory	3	40	60	**100**
Practical	1	40	60	**100**
Total	**4**	**200**		

Evaluation Scheme

	Marks	
	Internal	**External**
Theory	3 Internal Exams of 20 Marks (During the Semester) (Best 2 will be taken)	1 External Exams (At the End of the Semester)
Practical	3 Internal Exams (During the Semester) (Best 2 will be taken)	1 External Exams (At the End of the Semester)

Bachelor of Science (B.Sc.) **Subject: Computer Science**

III Semester **Paper: Elective, Computer Networks & Information Security**

Content of the Course

No. of Lectures (in hours per week): 2 Hrs. per week

Total No. of Lectures: 60 Hrs. **Maximum Marks:** 60

Units	Topics	No. of Lecture
I	Introduction to Computer Network: Use of Computer network: Access to information, person-to-person communication electronic commerce, internet of things. Types of computer networks: Broadband access network, Mobile and wireless network, content delivery network, transit network, Enterprise network. Network Technology: Personal Area Network Local Area Network, Metropolitan Area Network, Wide Area Network, example of network (Internet, Mobile network, wireless network); Reference Model: OSI, TCP/IP, Critique of the OSI and TCP/IP reference models.	10
II	Physical Layer: Guided Transmission Media: Twisted pairs, coaxial cable, Fiber Optics; Wireless transmission: The electromagnetic spectrum, frequency hopping spread spectrum, direct sequence, spread spectrum, ultra deb communication; Cellular Network: Common concepts-cells, handoff, 1G 2G,3G,4G & 5G technology.	10
III	Data Link Layer: Service Provided to Network Layer: Data Link Control: Framing, Flow and Error Control; Error detecting codes, Error-correcting codes; Data Link Protocols: Basic transmission and receipt, simplex, full duplex, sliding window protocol, Point-to-Point Protocol. Switching Techniques: Packet Switching, Circuit Switching, Datagram Networks, VirtualCircuit Networks. Network Devices & Drivers: Router, Modem, Repeater, Hub, Switch, Bridge and Gateway (fundamental concepts)	10
IV	Network Security and Information Security: Fundamentals of network and information security: principles of security and attack. Security Goals (Confidentiality, Integrity, and Availability). Overview of Security Threats and Vulnerability: Types of attacks on Confidentiality, Integrity and Availability. Vulnerability and Threats: Phishing Attacks, E-mail threats, web threats, Intruders and Hackers, Insider threats, SQL injection Attacks, Ransomware. Malware: Worms, Virus Spams, Adware, Spyware, Trojans.	15

References

Textbooks:

- Andrew S. J. Wetherall, Computer Networks, 6th Edition, (2021), Pearson.
- J Mattord, Principles of Information Security, Fourth Edition, 6th Indian Reprint.
- Praveen Kinnar Shur la, Surya Prakash Tripathi, Ritendra Goe 1 "Introduction to Information Security and Cyber Laws", 2014. Dreamtech Press.
- Books published by Hindi Granth Academy, Bhopal.

Reference Books:

- Kurose James F., Ross Keith W., Computer Networking, A 4 op-Down Approach, Sixth Edition, 2017. Pearson
- Micki Krausc. Harold F. Tipton, Handbook of Information Security Management, Vol. 1-3, CRC Press LLC.
- B. A. Forouzan: Data Communications and Networking. Fourth edition, TMH Publishing Company Ltd.
- Basta W. Halton, Computer Security: Concepts, Issues and Implementation, Cengage Learning India.

List of Practical

- Study of UTP cable
  - Color code of UTP cable Categories of UTP n/w cable
  - Shielding of n/w cable

  - Maximum length for which data cable can be usedCrimping of RJ45 connector and punching of datacab1e
- Knowledge of Structured Cabling and its componentsInformation
  Information outlet with box
  - Network Rack (4U, 6U, 9U, 12U, 24U, 32U, 42U)
  - Patch Panel
  - Rack Management
- Study of Optical Fiber cable
  - Different cores of OF C (6 core, 12, 24 core) Multimode & Single mode OFC cable Shielding ofOFC
  - Splicing/Termination of OFC. OTDR Testing
  - LIU fix
  - LIU management (pigtail/fiber patchcord) and MediaConverter.
  - FP module.
  - Rules of OFC laying
- Use of tools
  - Crimping tool
  - Punching tool
  - Nose plier
  - Wire stripping and cable cutter
  - Multi-meter
- Configuration/management of Local Area Network
  - Implementation of File and printer sharing Installationof ft server and client
  - Connect the computers to Local Area Network.
  - Configuring Class, A IP address on LAN Connection inComputer LAB and use the following tools:
    - Ping, ipconfig, getmac, hostname, nslookup,tracert, systeminfo.
    - routing using packet tracer software
    - Dynamic routing using packet tracer
    - Implementation of Subnetting in Class A, B, C
    - Ping between 2 s2'stems using lPv6

_____

Faculty of Science

Bachelor of Science (B.Sc.), III Semester

SUBJECT: COMPUTER SCIENCE

Paper-Elective

Computer Networks & Information Security

Course Outcomes

CO. No.	Course Outcomes	Cognitive Level
CO 1	Define and describe the components of a data communication system such as various protocols. OSI Model, data transmission in analog and digital format	U, A
CO 2	Identify and differentiate among the network devices and drives	K, An, Ap
CO 3	Learn and describe various error detection and correction methods. describe the Various terminologies used in the Network and Application layers.	U, Ap

Credit and Marking Scheme

	Credits	Marks		Total Marks
		Internal	External	
Theory	3	40	60	**100**
Practical	1	40	60	**100**
Total	**4**	**200**		

Evaluation Scheme

	Marks	
	Internal	**External**
Theory	3 Internal Exams of 20 Marks (During the Semester) (Best 2 will be taken)	1 External Exams (At the End of the Semester)
Practical	3 Internal Exams (During the Semester) (Best 2 will be taken)	1 External Exams (At the End of the Semester)

Content of the Course

Theory

No. of Lectures (in hours per week): 2 Hrs. per week

Total No. of Lectures: 60 Hrs. **Maximum Marks:** 60

Units	Topics	No. of Lecture
I	Introduction to Computer Network: Use of Computer network: Access to information, person-to-person communication electronic commerce, internet of things. Types of computer networks: Broadband access network, Mobile and wireless network, content delivery network, transit network, Enterprise network. Network Technology: Personal Area Network, Local Area Network, Metropolitan Area Network, Wide Area Network, example of network (Internet, Mobile network, wireless network); Reference Model: OSI, TCP/IP, Critique of the OSI and TCP/IP reference models.	10
II	Physical Layer: Guided Transmission Media: Twisted pairs, coaxial cable, Fiber Optics; Wireless transmission: The electromagnetic spectrum, frequency hopping spread spectrum, direct sequence, spread spectrum, ultra-deb communication; Cellular Network: Common concepts- cells, handoff, 1G 2G,3G,4G & 5G technology.	10
III	Data Link Layer: Service Provided to Network Layer: Data Link Control: Framing, Flow and Error Control; Error detecting codes, Error-correcting codes; Data Link Protocols: Basic transmission and receipt, simplex link layer protocol, full duplex, sliding window protocol, Packet over SONET, ADSL, Point-to-Point Protocol. Switching Techniques: Packet Switching, Circuit Switching, Datagram Networks, VirtualCircuit Networks, and Structure of a Switch. Network Devices & Drivers: Router, Modem, Repeater, Hub, Switch, Bridge and Gateway (fundamental concepts)	10
IV	Network Layer: Routing Algorithm: Optimality, Principle of Shortest Path Algorithm, Flooding, Distance Vector Routing, Broadcast Routing; Congestion in network, traffic management approaches; IP addresses, IPv4 Addresses, IPv6 Addresses. Network Security and Information Security: Fundamentals of network and information security: principles of security and attack. Security Goals (Confidentiality, Integrity, and Availability). Overview of Security Threats and Vulnerability: Types of attacks on Confidentiality, Integrity and Availability.	15

References

Textbooks:

- Andrew S. J. Wetherall, Computer Networks, 6th Edition,(2021), Pearson.
- J Mattord, Principles of Information Security, Fourth Edition, 6 th Indian Reprint.
  - Praveen Kinnar Shur la, Surya Prakash Tripathi, Ritendra Goe 1 "Introduction to Information Security an Cyber Laws", 2014. Dreamtech Press.
- Books published by Hindi Granth Academy, Bhopal

Reference Books:

- Kurose James F., Ross Keith W., Computer Networking, A 4 op-Down Approach, Sixth Edition, 2017. Pearson
- Micki Krausc. Harold F. Tipton, Handbook of Information Security Management, Vol. 1-3, CRC Press LLC.
- B. A. Forouzan: Data Communications and Networking. Fourth edition, TMH Publishing Company Ltd.
- Basta W. Halton, Computer Security: Concepts, Issues and Implementation, Cengage Learning India.

_____

Faculty of Science

Bachelor of Science (B.Sc.), IV Semester

SUBJECT: COMPUTER SCIENCE

Paper-Major/Minor

Object Oriented Programming with Java

Course Outcomes

CO. No.	Course Outcomes	Cognitive Level
CO 1	Implement Object Oriented programming concept using· basic syntaxes of control Structures, strings and functions for developing skills of logic building activity.	U, R
CO 2	Identify classes, objects, members of a class and the relationships among them needed for finding the solution to a specific problem.	R, An, U
CO 3	Demonstrates how to achieve re-usability using inheritance, interfaces and packages and describes how faster application development can be achieved	U, Ap, R
CO 4	Demonstrate, understanding and use of different exception-handling mechanisms and concepts of multi-threading for robust faster and efficient application development.	U, An, R
CO 5	Identify and describe a common abstract user interface. components to design GUI in Java using Applet & Graphics.	U, An
CO 6	Identify, Design & Develop complex Graphical user interfaces using AWT	U, Ap

Credit and Marking Scheme

	Credits	Marks		Total Marks
		Internal	External	
Theory	4	40	60	**100**
Practical	2	40	60	**100**
Total	**6**	**200**		

Evaluation Scheme

	Marks	
	Internal	**External**
Theory	3 Internal Exams of 20 Marks (During the Semester) (Best 2 will be taken)	1 External Exams (At the End of the Semester)
Practical	3 Internal Exams (During the Semester) (Best 2 will be taken)	1 External Exams (At the End of the Semester)

Bachelor of Science (B.Sc.) **Subject: Computer Science**

IV Semester **Paper: Major/Minor, Object-Oriented Programming with Java**

Content of the Course

No. of Lectures (in hours per week): 2 Hrs. per week

Total No. of Lectures: 60 Hrs. **Maximum Marks:** 60

Units	Topics	No. of Lecture
I	History, Java Features. How Java Differs from C and C++, Java and Internet, Java and World Wide Web, Java Supports Systems, Java Environment, lava Program Structure, Java Tokens. Constants, Variables, Scope of Variable, Data Types, Type Casting, Java Virtual Machine, Command Line, Arguments, Implementing a Java Program	10
II	Operators - Arithmetic Operators, Relational Operators, Logical Operators, Assignment Operators, Increment and Decrement Operators, Conditional Operators, Bitwise Operators, Special Operators, Arithmetic Expressions - Evaluation of Expressions, Precedence of Arithmetic Operators, Type Conversions in Expressions. Operator Precedence and Associativity, Mathematical functions. Decision making with if Statement, Simple if Statement, if. Else Statement. Nesting of if..else Statement, if-else Ladder, the Switch statement, The ?: Operator. Loops - While Statement, Do-while Statement, For Statement, Jump in Loops, Labeled Loops	15
III	Class - Defining a Class, Adding Variables, Adding Methods, Creating Objects, Accessing Class Members, Static Members, Methods- Defining Methods, Nesting of Methods. Method Overloading. Constructors: definition and types, Constructor Overloading. Inheritance - Extending a Class. Overloading Methods, Final Variables and Methods, Final Classes, Finalize Methods, Abstract Methods and Classes, Visibility Control. Arrays: One and two Dimensional Array, Strings. Vectors. Wrapper Class.	10
IV	Interface- Defining Interfaces, Extending Interfaces, Implementing Interfaces. Accessing Interface Variables. Packages - Using System Packages, Naming Conventions, Creating Packages, accessing a Package, Using a Package. Adding a Class to a Package. and hiding Classes. Multithreading Creating Threads. Extending the Thread Class, Life Cycle of a Thread. Implement the Runnable interface. Exceptions Handling: try, catch, finally.	15
V	Applets - building Applet Code, Applet Life Cycle, Applet Tag, Passing Parameters to Applets, Getting Input from the user. Applet Graphics Methods: drawstring, drawRect, fillRect, drawOval, fillOval, drawLine, drawImage, drawArc, fillArc, setColor, setFont. Concept of Stream - Stream C lasses, Byte Stream Classes, Character Stream Classes.	10

References

Text Books

- E Balguruswami, Programming with Java, Tata McGrnw-Hill Publication.

Reference Books

- Bruce Eckel, Thinking in Java.
- Herbert Schildt Java: The Complete Reference.
- Y. Daniel Liang, Introduction to Java Programming.
- Paul Deitel, Harvey Deitel, Java: How to Program.

Web Links:

- https://www.cs.cmu.edu/afs/cs.cniu.edu/usei/gclien/www/down1oad/java/LeainJava.pdf
- https://www. tutorialspoint.confjava/java tutorial.pdf
- httns://www.youtube.com/watch=7soxDfdgtDw

- http:ljwww.mphindigranthaca_gemy.org/
- Suggested equivalent on line courses : httns://nptel.ac.in/courses/106/105/106105191/

List of Practical

1. Find a greater number between two numbers -using a conditional operator.
2. Find the factorial of the number, the number is given by the user using the command line argument.
3. Write a program to check if a number is prime or not.
4. Write a program to display tables from 2 to 10.
5. Write a program to print the Fibonacci series.
6. Enter a no. and check whether it is even or odd.
7. Write a Program to find the sum & average of 10 no. using arrays.
8. Write a program to display the reverse of a digit no. using an array.
9. Write a program to demonstrate function overloading.
10. Write a program to display grades according to the marks obtained by the student.
11. Write a program to calculate the salary of an employee if the salary is greater than or equal to 20000 and the year of service is greater than or equal to 5 years then the bonus will be 2000 otherwise 1000 and print the grass salary of the employee.
12. Write a program to convert the given no. of days into months, and days using classes, objects and Methods.
13. Write a program to convert a given string into Uppercase and lowercase and get the length of the string using an array.
14. Create a package called "Arithmetic" that contains methods to deal all arithmetic operations. Also, write a program to use the package.
15. Write a program to demonstrate the use of a constructor and destructor.
16. Define an exception called the "Marks out of Bound" exception that is thrown if the entered marks are greater than 100.
17. Write a program using the application of single inheritance. Find the area of the rectangle & volume of a cube.
18. Develop a simple real-life application to illustrate the use of multithreading.
19. Write a program using multiple inheritances to calculate the area and perimeter of a circle using the interface.
20. Write an applet program to draw a Rectangle (color = orange) and a right-aligned oval.
21. Develop an applet that receives 3 numeric values as inputs from the user and then displays the largest no. on the screen.

Faculty of Science

Bachelor of Science (B.Sc.), IV Semester

SUBJECT: COMPUTER SCIENCE

Paper-Elective

Object Oriented Programming with Java

Course Outcomes

CO. No.	Course Outcomes	Cognitive Level
CO 1	Implement Object Oriented programming concept using· basic syntaxes of control Structures, strings and functions for developing skills of logic building activity.	U, R
CO 2	Identify classes, objects, members of a class and the relationships among them needed for finding the solution to a specific problem.	R, An, U
CO 3	Demonstrates how to achieve re-usability using inheritance, interfaces and packages and describes how faster application development can be achieved	U, Ap, R
CO 4	Demonstrate, understanding and use of different exception-handling mechanisms and concepts of multi-threading for robust faster and efficient application development.	U, An, R
CO 5	Identify and describe a common abstract user interface. components to design GUI in Java using Applet & Graphics.	U, An
CO 6	Identify, Design & Develop complex Graphical user interfaces using AWT	U, Ap

Credit and Marking Scheme

	Credits	Marks		Total Marks
		Internal	External	
Theory	3	40	60	**100**
Practical	1	40	60	**100**
Total	**4**	**200**		

Evaluation Scheme

	Marks	
	Internal	**External**
Theory	3 Internal Exams of 20 Marks (During the Semester) (Best 2 will be taken)	1 External Exams (At the End of the Semester)
Practical	3 Internal Exams (During the Semester) (Best 2 will be taken)	1 External Exams (At the End of the Semester)

Bachelor of Science (B.Sc.) **Subject: Computer Science**
IV Semester **Paper: Elective, Object-Oriented Programming with Java**

Content of the Course

No. of Lectures (in hours per week): 2 Hrs. per week

Total No. of Lectures: 60 Hrs. **Maximum Marks:** 60

Units	Topics	No. of Lecture
I	History, Java Features. How Java Differs from C and C++, Java and Internet, Java and World Wide Web, Java Supports Systems, Java Environment, lava Program Structure, Java Tokens. Constants, Variables, Scope of Variable, Data Types, Type Casting, Java Virtual Machine, Command Line, Arguments, Implementing a Java Program.	10
II	Operators - Arithmetic Operators, Relational Operators, Logical Operators, Assignment Operators, Increment and Decrement Operators, Conditional Operators, Bitwise Operators, Special Operators, Arithmetic Expressions - Evaluation of Expressions, Precedence of Arithmetic Operators, Type Conversions in Expressions. Operator Precedence and Associativity, Mathematical functions. Decision making with if Statement, Simple if Statement, if. Else Statement. Nesting of if..else Statement, if-else Ladder, the Switch statement, The ?: Operator. Loops - While Statement, Do-while Statement, For Statement, Jump in Loops, Labeled Loops	15
III	Class - Defining a Class, Adding Variables, Adding Methods, Creating Objects, Accessing Class Members, Static Members, Methods- Defining Methods, Nesting of Methods. Method Overloading. Constructors: definition and types, Constructor Overloading. Inheritance - Extending a Class. Overloading Methods, Final Variables and Methods, Final Classes, Finalize Methods, Abstract Methods and Classes, Visibility Control. Arrays: One and two Dimensional Array, Strings. Vectors. Wrapper Class.	10
IV	Interface- Defining Interfaces, Extending Interfaces, Implementing Interfaces. Accessing Interface Variables. Packages - Using System Packages, Naming Conventions, Creating Packages, accessing a Package, Using a Package. Adding a Class to a Package. and hiding Classes. Multithreading Creating Threads. Extending the Thread Class, Life Cycle of a Thread. Implement the Runnable interface. Exceptions Handling: try, catch, finally.	15

References

Text Books

- E Balguruswami, Programming with Java, Tata McGrnw-Hill Publication.

Reference Books

- Bruce Eckel, Thinking in Java.
- Herbert Schildt Java: The Complete Reference.
- Y. Daniel Liang, Introduction to Java Programming.
- Paul Deitel, Harvey Deitel, Java: How to Program.
- Cay S. Horsttnann, Core Java Volume I - Fundamentals.
- Java Projects, BPB Publication.
- Dr. S.S. Kandare, Programming in Java, S Chand Publication.
- Books published by M.P. Hindi Granth Academy, Bhopal.

Web Links:

- https://www.cs.cmu.edu/afs/cs.cniu.edu/usei/gclien/www/down1oad/java/LeainJava.pdf
- https://www. tutorialspoint.confjava/java tutorial.pdf
- httns://www.youtube.com/watch=7soxDfdgtDw

- http:ljwww.mphindigranthaca_gemy.org/
- Suggested equivalent on line courses : httns://nptel.ac.in/courses/106/105/106105191/

List of Practical

1. Find a greater number between two numbers -using a conditional operator.
2. Find the factorial of the number, the number is given by the user using the command line argument.
3. Write a program to check if a number is prime or not.
4. Write a program to display tables from 2 to 10.
5. Write a program to print the Fibonacci series.
6. Enter a no. and check whether it is even or odd.
7. Write a Program to find the sum & average of 10 no. using arrays.
8. Write a program to display the reverse of a digit no. using an array.
9. Write a program to demonstrate function overloading.
10. Write a program to display grades according to the marks obtained by the student.
11. Write a program to calculate the salary of an employee if the salary is greater than or equal to 20000 and the year of service is greater than or equal to 5 years then the bonus will be 2000 otherwise 1000 and print the grass salary of the employee.
12. Write a program to convert the given no. of days into months, and days using classes, objects and Methods.
13. Write a program to convert a given string into Uppercase and lowercase and get the length of the string using an array.
14. Create a package called "Arithmetic" that contains methods to deal all arithmetic operations. Also, write a program to use the package.
15. Write a program to demonstrate the use of a constructor and destructor.
16. Define an exception called the "Marks out of Bound" exception that is thrown if the entered marks are greater than 100.
17. Write a program using the application of single inheritance. Find the area of the rectangle & volume of a cube.
18. Develop a simple real-life application to illustrate the use of multithreading.
19. Write a program using multiple inheritances to calculate the area and perimeter of a circle using the interface.
20. Write an applet program to draw a Rectangle (color = orange) and a right-aligned oval.
21. Develop an applet that receives 3 numeric values as inputs from the user and then displays the largest no. on the screen.

Faculty of Science

Bachelor of Science (B.Sc.), IV Semester

SUBJECT: COMPUTER SCIENCE

Paper-Elective

Object Oriented Programming with Java

Course Outcomes

CO. No.	Course Outcomes	Cognitive Level
CO 1	Implement Object Oriented programming concept using· basic syntaxes of control Structures, strings and functions for developing skills of logic building activity.	U, R
CO 2	Identify classes, objects, members of a class and the relationships among them needed for finding the solution to a specific problem.	R, An, U
CO 3	Demonstrates how to achieve re-usability using inheritance, interfaces and packages and describes how faster application development can be achieved	U, Ap, R
CO 4	Demonstrate, understanding and use of different exception-handling mechanisms and concepts of multi-threading for robust faster and efficient application development.	U, An, R
CO 5	Identify and describe a common abstract user interface. components to design GUI in Java using Applet & Graphics.	U, An
CO 6	Identify, Design & Develop complex Graphical user interfaces using AWT	U, Ap

Credit and Marking Scheme

	Credits	Marks		Total Marks
		Internal	External	
Theory	3	40	60	**100**
Practical	1	40	60	**100**
Total	**4**	**200**		

Evaluation Scheme

	Marks	
	Internal	**External**
Theory	3 Internal Exams of 20 Marks (During the Semester) (Best 2 will be taken)	1 External Exams (At the End of the Semester)
Practical	3 Internal Exams (During the Semester) (Best 2 will be taken)	1 External Exams (At the End of the Semester)

Content of the Course

Theory

No. of Lectures (in hours per week): 2 Hrs. per week

Total No. of Lectures: 60 Hrs. **Maximum Marks:** 60

Units	Topics	No. of Lecture
I	History, Java Features. How Java Differs from C and C++, Java and Internet, Java and World Wide Web, Java Supports Systems, Java Environment, lava Program Structure, Java Tokens. Constants, Variables, Scope of Variable, Data Types, Type Casting, Java Virtual Machine, Command Line, Arguments, Implementing a Java Program	10
II	Operators - Arithmetic Operators, Relational Operators, Logical Operators, Assignment Operators, Increment and Decrement Operators, Conditional Operators, Bitwise Operators, Special Operators, Arithmetic Expressions - Evaluation of Expressions, Precedence of Arithmetic Operators, Type Conversions in Expressions. Operator Precedence and Associativity, Mathematical functions. Decision making with if Statement, Simple if Statement, if. Else Statement. Nesting of if..else Statement, if-else Ladder, the Switch statement, The ?: Operator. Loops - While Statement, Do-while Statement, For Statement, Jump in Loops, Labeled Loops	15
III	Class - Defining a Class, Adding Variables, Adding Methods, Creating Objects, Accessing Class Members, Static Members, Methods- Defining Methods, Nesting of Methods. Method Overloading. Constructors: definition and types, Constructor Overloading. Inheritance - Extending a Class. Overloading Methods, Final Variables and Methods, Final Classes, Finalize Methods, Abstract Methods and Classes, Visibility Control. Arrays: One and two Dimensional Array, Strings. Vectors. Wrapper Class.	10
IV	Interface- Defining Interfaces, Extending Interfaces, Implementing Interfaces. Accessing Interface Variables. Packages - Using System Packages, Naming Conventions, Creating Packages, accessing a Package, Using a Package. Adding a Class to a Package. and hiding Classes. Multithreading Creating Threads. Extending the Thread Class, Life Cycle of a Thread. Implement the Runnable interface. Exceptions Handling: try, catch, finally.	15

References

Text Books

- E Balguruswami, Programming with Java, Tata McGrnw-Hill Publication.

Reference Books

- Bruce Eckel, Thinking in Java.
- Herbert Schildt Java: The Complete Reference.
- Y. Daniel Liang, Introduction to Java Programming.
- Paul Deitel, Harvey Deitel, Java: How to Program.
- Cay S. Horsttnann, Core Java Volume I - Fundamentals.
- Java Projects, BPB Publication.
- Dr. S.S. Kandare, Programming in Java, S Chand Publication.
- Books published by M.P. Hindi Granth Academy, Bhopal.

Web Links:

- https://www.cs.cmu.edu/afs/cs.cniu.edu/usei/gclien/www/down1oad/java/LeainJava.pdf
- https://www. tutorialspoint.confjava/java tutorial.pdf
- httns://www.youtube.com/watch=7soxDfdgtDw

- http:ljwww.mphindigranthaca_gemy.org/
- Suggested equivalent on line courses : httns://nptel.ac.in/courses/106/105/106105191/

List of Practical

22. Find a greater number between two numbers -using a conditional operator.
23. Find the factorial of the number, the number is given by the user using the command line argument.
24. Write a program to check if a number is prime or not.
25. Write a program to display tables from 2 to I 0.
26. Write a program to print the Fibonacci series.
27. Enter a no. and check whether it is even or odd.
28. Write a Program to find the sum & average of 10 no. using arrays.
29. Write a program to display the reverse of a digit no. using an array.
30. Write a program to demonstrate function overloading.
31. Write a program to display grades according to the marks obtained by the student.
32. Write a program to calculate the salary of an employee if the salary is greater than or equal to 20000 and the year of service is greater than or equal to 5 years then the bonus will be 2000 otherwise 1000 and print the grass salary of the employee.
33. Write a program to convert the given no. of days into months, and days using classes, objects and Methods.
34. Write a program to convert a given string into Uppercase and lowercase and get the length of the string using an array.
35. Create a package called "Arithmetic" that contains methods to deal all arithmetic operations. Also, write a program to use the package.
36. Write a program to demonstrate the use of a constructor and destructor.
37. Define an exception called the "Marks out of Bound" exception that is thrown if the entered marks are greater than 100.
38. Write a program using the application of single inheritance. Find the area of the rectangle & volume of a cube.
39. Develop a simple real-life application to illustrate the use of multithreading.
40. Write a program using multiple inheritances to calculate the area and perimeter of a circle using the interface.
41. Write an applet program to draw a Rectangle (color = orange) and a right-aligned oval.
42. Develop an applet that receives 3 numeric values as inputs from the user and then displays the largest no. on the screen.
43. Write a Java Program to read data from the inputted text file name, and print its content on the console.
44. Write a Java Program to merge two files into a third file
45. Write a Java program to delete duplicate lines in a text file
46. Write a Java Program to implement FilelnputStream class to read binary data from any image file.

___

Faculty of Science

Bachelor of Science (B.Sc.) V Semester

SUBJECT: COMPUTER SCIENCE

Paper-Core

Relational Database Management System

CO. No.	Course Outcomes	Cognitive Level
CO 1	Understand database concepts, applications, structure, need for a database and ER-Model terminologies.	U, A
CO 2	Be able to understand the fundamentals of Relational Algebra and relational calculus	K, U
CO 3	To gain skills in creating a logical design of databases, including the E R method and normalization approach.	U
CO 4	Know about SQL functions and operators	U, A
CO 5	Understand the knowledge of Database and transaction management.	U

Credit and Marking Scheme

	Credits	Marks		Total Marks
		Internal	External	
Theory	4	40	60	**100**
Practical	2	40	60	**100**
Total	**6**	**200**		

Evaluation Scheme

	Marks	
	Internal	**External**
Theory	3 Internal Exams of 20 Marks (During the Semester) (Best 2 will be taken)	1 External Exams (At the End of the Semester)
Practical	3 Internal Exams (During the Semester) (Best 2 will be taken)	1 External Exams (At the End of the Semester)

Bachelor of Science (B.Sc.) **Subject: Computer Science**

V Semester **Paper: Core, Relational Database Management System**

Content of the Course

No. of Lectures (in hours per week): 2 Hrs. per week

Total No. of Lectures: 60 Hrs. **Maximum Marks:** 60

Units	Topics	No. of Lectures
I	Introduction: Advantages of the DBMS approach, various views of data, data independence, schema & sub-schema, Primary concepts of data models, Database languages, transaction management, database administrator, & uses, data dictionary, and overall system architecture. ER Model: Basic concepts, design issues, mapping constraints, keys, ER diagram, weak & strong entity sets, specialization & generalization, aggregation, inheritance, design of ER schema, reduction of ER schema to tables.	12
II	Domain Relation & Keys: Domains, relations, kinds of relation, relational databases, various types of keys, candidate, primary, alternate & foreign keys. Relation algebra & SQL: The structure, relation algebra with extended operations, modification of database, idea of relational calculus, basic structure of SQL, set operation, aggregate function.	12
III	Functional dependencies & Normalization: Base definition, trivial and nontrivial dependencies, closure set of dependencies, & of attributes, irreducible set of dependencies, introduction to normalization, non-loss decomposition, FD diagram, I,II & III NF, dependencies prevention, BCNF, multivalued dependencies, preventions, BCNF, Multivalued dependencies & 4NF, Join dependencies & 5NF.	12
IV	Introduction to SQL, Data types, key constraints:- primary key, Candidate key, Integrity rules Entity integrity, Referential integrity rule. SQL Commands: - DDL, DML, DCL, TCL syntax and examples, select query with all the clauses. Like Predicate, Operator (Between, In, Not in)	12
V	Advanced SQL: - SQL join operations, Sub queries, indexes, sequences, and views SQL Functions. Introduction to PL/SQL:-PL/SQL structure, Cursors, Triggers, Stored Procedures and functions. Transaction Management-concurrency & recovery, ACID properties, transaction state, implementation of atomicity and durability, Storage structure in database - types, hashing.	12

References

Text Books:

- Database concepts by Henry F. Korth, MGH
- An Introduction to Database System by Bipin C. Desai, Galgotia Pub.

Reference Books:

- Database Management system by Arun K. Majumdar & P. Bhattacharya, TMH Pub.
- Principles of Database System by Jeffrey O. Ullman, Galgotia Pub, Co. Ltd.
- Principles of Database Management System by James Martin, PHI

List of Practical

Sample Table – Worker

WORKER_ID	FIRST_NAME	LAST_NAME	SALARY	JOINING_DATE	DEPARTMENT

Sample Table – Bonus

WORKER_REF_ID	BONUS_DATE	BONUS_AMOUNT

Sample Table – Title

WORKER_REF_ID	WORKER_TITLE	AFFECTED_FROM

1. Write an SQL query to fetch "FIRST_NAME" from the Worker table using the alias name as <WORKER_NAME>.
2. Write an SQL query to fetch "FIRST_NAME" from the Worker table in upper case.
3. Write an SQL query to fetch unique values of DEPARTMENT from the Worker table.
4. Write an SQL query to print the first three characters of FIRST_NAME from the Worker table.
5. Write an SQL query to find the position of the alphabet ('a') in the first name column 'Amitabh' from the Worker table.
6. Write an SQL query to print the FIRST_NAME from the Worker table after removing white spaces from the right side.
7. Write an SQL query to print the DEPARTMENT from the Worker table after removing white spaces from the left side.
8. Write an SQL query that fetches the unique values of DEPARTMENT from the Worker table and prints its length.
9. Write an SQL query to print the FIRST_NAME from the Worker table after replacing 'a' with 'A'.
10. Write an SQL query to print the FIRST_NAME and LAST_NAME from the Worker table into a single column COMPLETE_NAME. A space char should separate them.
11. Write an SQL query to print all Worker details from the Worker table order by FIRST_NAME Ascending.
12. Write an SQL query to print all Worker details from the Worker table order by FIRST_NAME Ascending and DEPARTMENT Descending.
13. Write an SQL query to print details for Workers with the first names as "Vipul" and "Satish" from the Worker table.
14. Write an SQL query to print details of Workers with DEPARTMENT name as "Admin".
15. Write an SQL query to print details of the Workers whose FIRST_NAME contains 'a'.
16. Write an SQL query to print details of the Workers whose FIRST_NAME ends with 'a'.
17. Write an SQL query to print details of the Workers whose FIRST_NAME ends with 'h' and contains six alphabets.
18. Write an SQL query to print details of the Workers whose SALARY lies between 100000 and 500000.
19. Write an SQL query to print details of the Workers who joined in Feb'2014. Write an SQL query to fetch the count of employees working in the department 'Admin'.
20. Write an SQL query to fetch worker names with salaries >= 50000 and <= 100000.
21. Write an SQL query to fetch the no. of workers for each department in descending order.
22. Write an SQL query to print details of the Workers who are also Managers.
23. Write an SQL query to fetch duplicate records having matching data in some fields of a table.
24. Write an SQL query to show only odd rows from a table.
25. Write an SQL query to show only even rows from a table.
26. Write an SQL query to clone a new table from another table.
27. Write an SQL query to fetch intersecting records of two tables.
28. Write an SQL query to show records from one table that another table does not have.

Faculty of Science

Bachelor of Science (B.Sc.) V Semester

SUBJECT: Computer Science

Paper-DSE I

SOFTWARE ENGINEERING

Course Outcomes

CO. No.	Course Outcomes	Cognitive Level
CO 1	Learn and understand the Concepts of Software Engineering	U
CO 2	Learn and understand the Software Development Life Cycle	U, K, A
CO 3	To Understand and apply the analysis principles to software development.	U, A
CO 4	To apply the design & testing principles to software engineering.	U, A, K

Credit and Marking Scheme

	Credits	Marks		Total Marks
		Internal	External	
Theory	4	40	60	**100**
Total	**4**	**100**		

Evaluation Scheme

	Marks	
	Internal	External
Theory	3 Internal Exams of 20 Marks (During the Semester) (Best 2 will be taken)	1 External Exams (At the End of the Semester)

Bachelor of Science (B.Sc.) **Subject: Computer Science**

V Semester **Paper: DSE-I, Software Engineering System**

No. of Lectures (in hours per week): 3 Hrs. per week

Total No. of Lectures: 60 Hrs. **Maximum Marks:** 100

Units	Topics	No. of Lectures
I	**Software-**What is software? Types of software, Characteristics of Software, Attributes of good software, **Software Engineering-**What is software engineering? Software engineering costs, what are the key challenges facing software engineering? Systems engineering & Software Engineering.	12
II	**Software Development Process Models**-What is a software process? What is a software process model? The waterfall model, Evolutionary development, Component-Based Software Engineering (CBSE). **Process Iteration**-Incremental delivery, Spiral development. **Rapid software development**-Agile methods, Extreme programming, Rapid application development (RAD), Software prototyping. **Computer Aided Software Engineering (CASE)-**Overview of CASE approach, Classification of CASE tools.	12
III	**Software Requirement Analysis and Specification**-System and software requirements, Types of software requirements- Functional and non-functional requirements, Domain requirements, User requirements. **Elicitation and analysis of requirements**- Overview of techniques, Viewpoints Interviewing, Scenarios, Use-cases, Process modeling with physical and logical DFDs, Entity Relationship Diagram, Data Dictionary, Requirement validation, Requirement specification, Software requirement Specification (SRS), Structure and contents of SRS, SRS format. **Software Size Estimation and Cost Estimation**-Software Estimation –Size Estimation, Function Point Analysis, LOC Estimation, COCOMO.	12
IV	**Software Design**-Design Concepts-Abstraction, Architecture, Patterns, Modularity, Cohesion Coupling, Information hiding, Functional independence, Refinement. Design of input and Control, Design of User Interface design- Elements of good design, Design issues. Features of modern GUI - Menus, Scroll bars, windows, Buttons, icons, panels, error Messages, etc.	12
V	Good programming practices and Coding Standards. **Software Testing and Quality Assurance**-Verification and validation, Techniques of testing-Black-box and White-box testing, **Levels of testing**-Unit testing, Integration Testing, Interface testing, System testing, Alpha and beta testing, Regression testing. Design of test cases, Quality management activities, Product and process quality, Standards-ISO9000, Capability Maturity Model (CMM).	12

References

- Software Engineering, A practitioner's Approach- Roger S. Pressman, 6th edition, Mc Graw Hill International Edition.
- An Integrated Approach to Software Engineering by Pankaj Jalote.
- Software Engineering- Sommerville, 7th edition, Pearson Education.
- The unified modeling language user guide Grady Booch, James Rambaugh, Ivar Jacobson, Pearson Education.

Reference Books:

- Software Engineering, an Engineering approach- James F. Peters, Witold Pedrycz, John Wiley.
- Software Engineering principles and practice- Waman S Jawadekar, The McGraw-Hill Companies.
- Fundamentals of object-oriented design using UML Meiler page-Jones: Pearson Education.

Mrs S. J

Faculty of Science

Bachelor of Science (B.Sc.), V Semester

SUBJECT: COMPUTER SCIENCE

Paper-Elective (DSE-II)

Operating System

Course Outcomes

CO. No.	Course Outcomes	Cognitive Level
CO 1	Describe the importance of computer system resources and the role of operating systems in their management policies and algorithms.	U, Ap
CO 2	Specify the objectives of modern operating systems and describe how operating systems have evolved over time	K, An
CO 3	Understand various process management concepts and can compare various scheduling techniques, synchronization, and deadlocks.	U
CO 4	Describe the concepts of multithreading and memory management techniques.	U, An
CO 5	Identify the best-suited memory management techniques.	U

Credit and Marking Scheme

	Credits	Marks		Total Marks
		Internal	External	
Theory	3	40	60	**100**
Practical	1	40	60	**100**
Total	**4**	**200**		

Evaluation Scheme

	Marks	
	Internal	**External**
Theory	3 Internal Exams of 20 Marks (During the Semester) (Best 2 will be taken)	1 External Exams (At the End of the Semester)
Practical	3 Internal Exams (During the Semester) (Best 2 will be taken)	1 External Exams (At the End of the Semester)

Bachelor of Science (B.Sc.)
V Semester

Subject: Computer Science
Paper: DSE-II, Operating System

Content of the Course

No. of Lectures (in hours per week): 2 Hrs. per week

Total No. of Lectures: 60 Hrs. **Maximum Marks:** 60

Units	Topics	No. of Lectures
I	Introduction to Operating System: What is an Operating System? History and Evolution of OS, Basic OS functions, Resource Abstraction, Types of Operating Systems- Multiprogramming Systems, Batch Systems, Time Sharing Systems; Operating Systems for Personal Computers, Workstations and Hand-held Devices, Process Control & Real-time Systems.	12
II	Process Management: Process Concepts, Process States & Process Control Block. Process Scheduling: Scheduling Criteria, Scheduling Algorithms (Preemptive & Non-Preemptive) - FCFS, SJF, SRTN, RR, Priority, Multiple-Processor, Real-Time, Multilevel Queue and Multilevel Feedback Queue Scheduling. Deadlock - Definition, Deadlock Characterization, Necessary and Sufficient Conditions for Deadlock. Deadlock Handling Approaches: Prevention, Avoidance, Detection and Recovery.	12
III	Memory Management: Introduction, Address Binding, Logical versus Physical Address Space, Swapping, Contiguous & Non-Contiguous Allocation, Fragmentation (Internal & External), Compaction, Paging, Segmentation, Virtual Memory, Demand Paging, Performance of Demand Paging, Page Replacement Algorithms. File Management: Concept of File System (File Attributes, Operations, Types), Functions of File System, Types of File System, Access Methods (Sequential, Direct & other methods),	12
IV	Disk Management: Structure, Disk Scheduling Algorithms (FCFS, SSTF, SCAN, C- SCAN, LOOK),LINUX: Introduction, History and features of Linux, advantages, Linux architecture, the File system of LINUX - boot block, super block, inode table, data blocks. Linux standard directories, Linux kernel, Partitioning the hard drive for Linux, system - startup and shut-down process, init and run levels. Process, Swap, Partition, fdisk, checking disk free spaces. CLI OS vs GUI OS, Windows v/s Linux, Files and Directories. Concept of Open-Source Software.	12
V	Linux Administration: Types of user- Root and normal user, Multiple logins simultaneously (Ctrl + Alt + F1, F2,F6), who command. Help: what is, --help, man command Basic Commands: For displaying current directory, files and directories of current/absolute/relative location(s), creating, removing, renaming, copying and moving files or directories.Managing multiple processes: connecting processes with pipes, tee, redirecting input-output, changing process priority with nice, cron commands, kill, ps. Managing user accounts- Sudo, users: useradd, usermod, userdel, passwd. Group: Primary & Secondary Group, chgrp, chown, groupadd, groupdel. Permissions: adding and removing permissions.	12

References

Text Books:

- A Silberschatz , P.B Galvin , G. Gagne , Operating System Concepts, John Wiley Publications
- A.S Tanenbaum, Modern Operating System, Pearson Education
- J.L Peterson, Operating System Concepts Sumitabh Das, Linux, TMH

Reference Books:

- G.Nutt, Operating Systems: A Modern Perspective, Pearson Education
- W. Stallings, Operating Systems, Internals & Design Principles, Pearson Education
- M.Milenkovic, Operating Systems – Concepts and Design, Tata McGraw Hill

Web Links:

- https://nptel.ac.in/courses/106/102/106102132/
- **https://web.iitd.ac.in/~mythili/os/**

- https://www.youtube.com/watch?v=aCJ3YgoolHQ

List of Practical

1. Linux Directory Commands: pwd, mkdir, rm -rf, ls, cd, cd/, cd-
2. Linux File Commands: touch, cat, cat >, cat >>, rm, cp, mv, rename
3. Linux Permission Commands: su, id, useradd, passwd, groupadd, chmod, groupdel, chown, chgrp
4. Linux File Content & Filter Commands: head, tail, tac, more, less, grep, cat, cut, grep, comm, sed, tee, tr, uniq, wc, od, sort, diff
5. Linux Utility Commands: find, bc, locate, date, cal, sleep, time, df, mount, exit, clear, gzip, gunzip
6. Linux Networking Commands: ip, ssh, mail, ping, host
7. Edit Crontab file: to wall message on the system at a particular time automatically.
8. Vi editor: Create a file, edit, save and quit. Highlighting the searched term within a file, cut, yank, undo.
9. Write the Linux command to copy a File to some other location.
10. Write the Linux command to delete a file.

Mr. P B

Faculty of Science

Bachelor of Science (B.Sc.) VI Semester

SUBJECT: COMPUTER SCIENCE

Paper-Core

Data Science using Python

Course Outcomes

CO. No.	Course Outcomes	Cognitive Level
CO 1	Understand database concepts, applications, structure, the need for a database, and ER-Model terminologies.	U, A
CO 2	Be able to understand the fundamentals of Relational Algebra and relational calculus	K,U
CO 3	To gain skills in creating the logical design of databases, including the E R method and normalization approach.	U
CO 4	Know about SQL functions and operators	U, A
CO 5	Understand the knowledge of Database and transaction management.	U

Credit and Marking Scheme

	Credits	Marks		Total Marks
		Internal	External	
Theory	4	40	60	**100**
Practical	2	40	60	**100**
Total	**6**	**200**		

Evaluation Scheme

	Marks	
	Internal	**External**
Theory	3 Internal Exams of 20 Marks (During the Semester) (Best 2 will be taken)	1 External Exams (At the End of the Semester)
Practical	3 Internal Exams (During the Semester) (Best 2 will be taken)	1 External Exams (At the End of the Semester)

Content of the Course

Theory

No. of Lectures (in hours per week): 2 Hrs. per week

Total No. of Lectures: 60 Hrs. **Maximum Marks:** 60

Units	Topics	No. of Lecture
I	Introduction to Data Science, Data Science and Artificial Intelligence, Basic concept of data mining: KDD process and its steps, Types of data in data science, Data Science functionalities: Introduction to ML and its types Preparing Data, Data Preprocessing and its Needs, Data Cleaning, Data Integration and Transformation, Data Normalization, Data Reduction, Principal Component Analysis (PCA), Discretization and Concept Hierarchy Generation.	12
II	Association Rule Mining: Basic Concept of Support and Confidence, Apriori Algorithms for mining frequent item-sets. Improving the accuracy of Apriori Algorithm FP-Growth algorithm. Classification and Prediction: Decision Tree Induction, Bayesian Classification, KNN, And Classification by Backpropagation: Neural Network, ANN and SVM, Classifier Accuracy.	12
III	Cluster Analysis: Introduction, Partitioning Methods, Hierarchical Methods, Density-Based Methods, Outlier Analysis. Cluster Model accuracy. Python for Data Science, Data types in Python, Input-output statements in Python, Control statements in Python, arrays and functions in Python. Operators in Python	12
IV	List and its operation, Dictionary, Data analysis using Python- pandas, importing and reading a CSV sheet, basic exploration of data, converting a Python data structure to a data frame, numerical description of a data frame, understanding iloc() and loc(), tackling Null values, data frames(concatenating, merging, join), Binning with Pandas	12
V	Numpy, Data frames, Data visualization using Matplotlib, Python libraries for Data Science, scilab, scikit, tensor flow, Information extraction using NLP, NLTK library using NLP, Case study Sentimental Analysis, Reinforcement Learning. Case study: Prediction of the disease in health services by building a model.	12

References

Textbooks:

- Data Mining – Concepts and Techniques - Jiawei Han & Micheline Kamber Harcourt, India.
- Eric Matthes, Python Crash Course: A Hands-On, Project-Based Introduction to Programming (2nd Edition)
- Zed A. ShawLearn Python the Hard Way: 3rd Edition
- John M. ZellePython Programming: An Introduction to Computer Science (3rd Edition)

List of Practical
Max Marks = 100 (60 External+40 Internal)

1. Create a list of random numbers and classify them as discrete or continuous variables.
2. Convert a numerical variable into a categorical variable based on specific criteria.
3. Calculate the mean, mode, median, standard deviation, variance, covariance, and correlation of a given dataset.
4. Perform a regression analysis to determine the relationship between two numerical variables.
5. Use the pandas library to read a CSV file using the read_csv() function.
6. Use functions like head(), tail(), info(), and describe() to get an overview of the data.
7. Convert a Python list, dictionary, or NumPy array to a DataFrame using the pandas library.
8. Calculate statistical measures like mean, median, and standard deviation on DataFrame columns.
9. Use iloc() for integer-based indexing and loc() for label-based indexing to access specific rows or columns in a data frame.
10. Identify and handle missing or Null values using functions like isnull(), fillna(), or dropna().
11. Perform DataFrame operations like concatenating, merging, and joining multiple DataFrames using concat(), merge(), and join() functions.
12. Use NumPy functions for indexing, reshaping arrays, generating random values, and performing mathematical operations on arrays.

Faculty of Science

Bachelor of Science (B.Sc.), VI Semester

SUBJECT: COMPUTER SCIENCE

Paper: DSE-IA

Subject: PHP and MySQL

Course Outcomes

CO. No.	Course Outcomes	Cognitive Level
CO 1	Develop a strong foundation in HTML, CSS, and JavaScript, enabling students to create and style dynamic web pages with text, links, tables, images, forms, and interactive elements.	U, R
CO 2	Gain proficiency in PHP, including its history, characteristics, installation, configuration, and language basics. Students can handle data types, variables, expressions, operators, control flow, and string manipulation effectively.	U, R, Ap
CO 3	Acquire skills in working with complex data structures such as arrays and multidimensional arrays. Understand and apply functions, object-oriented programming concepts (classes, objects, inheritance, polymorphism), and develop the ability to handle file operations and database access using PHP.	U, Ap, C
CO 4	Learn to set up web pages with PHP to handle various form elements and master file operations, including creating, reading, writing, renaming, deleting files, and managing file information.	An, Ap, C
CO 5	Master PHP for database access, including connecting to MySQL, creating databases and tables, and performing essential operations like inserting, updating, deleting, and selecting data.	U, Ap, C

Credit and Marking Scheme

	Credits	Marks		Total Marks
		Internal	External	
Theory	4	40	60	**100**
Practical	2	40	60	**100**
Total	**6**	**200**		

Evaluation Scheme

	Marks	
	Internal	**External**
Theory	3 Internal Exams of 20 Marks (During the Semester) (Best 2 will be taken)	1 External Exams (At the End of the Semester)
Practical	3 Internal Exams (During the Semester) (Best 2 will be taken)	1 External Exams (At the End of the Semester)

Bachelor of Science (B.Sc.)
VI Semester

Subject: Computer Science
Paper: DSE-IA, PHP and MySQL

Content of the Course

No. of Lectures (in hours per week): 2 Hrs. per week

Total No. of Lectures: 60 Hrs. **Maximum Marks:** 60

Units	Topics	No. of Lectures
I	Overview of HTML, Working with Text, Link, Table, Image, Forms, Input. Introduction of cascading style sheet, selector, inline, internal, external CSS, CSS in text, image. Overview of JavaScript, Variables, Operators, Control flow statements, Popup Boxes, Functions, Events, Windows and Document Objects, Array.	10
II	A Brief History of PHP, PHP Characteristics, Installing and Configuring PHP on Windows, PHP Language Basics: Lexical Structure, Data Types, Variables, Expressions and Operators, Decision Statements, Flow Control Statements, Embedding PHP in Web Pages. Strings: String Constants, Printing Strings, Accessing Individual Characters, String Handling Functions: length, Word count, string position, reverse, replace.	10
III	Arrays: Indexed Arrays, Associative Arrays, Identifying Elements of an Array, Storing Data in Arrays, Multidimensional Arrays, extracting multiple values, converting between arrays and variables, Traversing Arrays, Sorting. Functions: Calling a Function, defining a Function, Variable Scope, Function Parameters, Return Values, Variable Functions, Anonymous Functions. Object Oriented Programming Concepts: Classes, Objects, Member Functions, Encapsulations, Inheritance, and Polymorphism.	10
IV	Form Handling in PHP: Setting Up Web Pages to Communicate with PHP, Handling Text Fields, Text Areas, Check Boxes, Radio Buttons, List Boxes, Password Controls, Hidden Controls, Image Maps. File Handling: Working with files and directories, File Open and Read, File Create and Write, Reading and writing Character in file, reading entire file, Rename and Delete File, getting Information of files, ownership and permissions.	15
V	Database Access: Using PHP to access a database. Introduction to MySql, Connect and create database, create tables, insert, update, delete, select.	15

References

Text Books:

- Programming PHP by Rasmus Lerdorf and Kevin Tatroe, O'Reilly Publications
- Beginning PHP5 by Wrox Publication
- HTML 5, Black Book by DreamTech Press

Reference Books:

- Mastering PHP: BPB Publication
- PHP 5.1 for beginners by Evan Bayross and Sharman Shah, SPD Publications
- PHP 5.2 The Complete Reference by Steven Holzner, McGraw Hill Edition 2008.

Suggestive digital platforms/ web links:

- https://www.w3schools.com/php/
- https://www.learn-php.org/
- https://www.javatpoint.com/php-tutorial Part D-Assessment

List of Practical

1. Write HTML codes for displaying images and demonstrate hyperlinking.
2. Create a Feedback Form Using Form Handling.
3. Write a code for design menu system using list tag.
4. Apply CSS formatting to create page.
5. Write a PHP script to display a Welcome message.
6. Write a PHP script to demonstrate use of arithmetic operators, comparison operators, and logical operators.
7. Write a PHP script to set the type of variable using type casting.
8. Write PHP Script to print the Fibonacci series.
9. Write PHP Script to generate results and display grades.
10. Write PHP Script to find the maximum number out of three given numbers.
11. Write PHP Script using two-dimensional arrays such as the addition of two 2×2 matrices.
12. Write PHP Script for FOR EACH loop execution.
13. Write PHP script Using the user-defined function.
14. Write PHP script to demonstrate use of string function.
15. Write PHP script to demonstrate use of date/time function and Math functions.
16. Write a program to read input data, from table and display all this information in tabular form on output screen.
17. Write a program to manipulate data and display all this information in table format.
18. Create form to search data.
19. Develop small PHP application(s) using forms and database with updated and delete options.
20. Open and Read a file.

Faculty of Science

Bachelor of Science (B.Sc.), VI Semester

SUBJECT: COMPUTER SCIENCE

Paper- DSE IB

Cloud Computing

Course Outcomes

CO. No.	Course Outcomes	Cognitive Level
CO 1	Know the fundamentals of Cloud, Cloud Architecture, and types of services and Deployment Models.	U, R
CO 2	Understand the concept of virtualization and how this has enabled the development of Cloud Computing	R, U, Ap
CO 3	Understand Big Data technologies, HADOOP, Cloud databases and cloud security	U, R
CO 4	Explore some important cloud computing Open Source and Commercial Clouds Platforms.	U, An
CO 5	Solve a real-world problem using cloud computing through group collaboration.	U,An, Ap

Credit and Marking Scheme

	Credits	Marks		Total Marks
		Internal	External	
Theory	4	40	60	**100**
Practical	2	40	60	**100**
Total	**6**	**200**		

Evaluation Scheme

	Marks	
	Internal	**External**
Theory	3 Internal Exams of 20 Marks (During the Semester) (Best 2 will be taken)	1 External Exams (At the End of the Semester)
Practical	3 Internal Exams (During the Semester) (Best 2 will be taken)	1 External Exams (At the End of the Semester)

Bachelor of Science (B.Sc.)
VI Semester

Subject: Computer Science
Paper: DSE-IB, Cloud Computing and Big Data

No. of Lectures (in hours per week): 2 Hrs. per week

Total No. of Lectures: 60 Hrs. **Maximum Marks:** 60

Units	Topics	No. of Lecture
I	Introduction to Cloud Computing: Overview, NIST Definition of Cloud Computing, NIST features, Historical Development, Need for Cloud Computing, Principles of Cloud Computing, Challenges and Risk of Cloud Computing, Advantages and Drawbacks. Cloud applications: Scientific Applications, Business and Consumer Applications, Productivity Applications, Online Social Networking Applications, and Media Applications.	12
II	Cloud Architecture and Model: Cloud Computing Architecture, Cloud Reference Model, NIST Cloud Computing Reference Architecture, IBM Cloud Computing Reference Architecture, Cloud Security Alliance (CSA) Reference Model. Service and Deployment Models: Infrastructure As A Service (LaaS), Platform As A Service (PaaS), Software As A Service (SaaS), Other Cloud Service Models, Public, Private, community and Hybrid Deployment Models.	12
III	Virtualization Technology: Meaning and Level of Virtualization, Hypervisor, Type 1 Hypervisor, Type 2 Hypervisors. Full virtualization, Para-virtualization, and Hardware Assisted Virtualization. Types of Virtualizations: Server Virtualization, Storage Virtualization, Network Virtualization, Operating System Virtualization, Application Server Virtualization, Creating A Virtual Machine.	12
IV	Big Data: Overview, Need of Big Data, Characteristics, Benefits of Big Data Processing, Big Data Technologies, HADOOP: Hadoop Architecture, Hadoop Ecosystem, HDFS Architecture. MapReduce. Cloud Database NoSQL: Relational, non-relational vs. DBaaS Cloud Database, Cloud Databases, Amazon Dynamo Database, HBase, Cassandra, MongoDB.	12
V	Cloud Security: Cloud Information Security Fundamentals, Cloud Security Services, Cloud Security Concerns, Security Challenges, Infrastructure Security, Cloud computing security architecture. Case Study Open-Source Clouds Platform: Hadoop, OpenStack, Cloud Stack, Eucalyptus, OpenNebula. Commercial Clouds Platform: Google App Engine, Microsoft Azure, Amazon	12

References

Text Books:

- Cloud Computing by Pawan Thakur, Susheela Pathania (Satya Prakashan New Delhi)
- Cloud Computing By Gautam Shroff (Cambridge Enterprise)
- Cloud Computing: A Practical Approach by Antohy T Velte (McGraw Hill)
- Cloud Computing Bible by Barrie Sosinsky(Wiley India)

Reference Books:

- Bloor R., Kaufman M., Halper F. Judith Hurwitz "Cloud Computing for Dummies", Wiley India Edition.
- John Rittinghouse & James Ransome, "Cloud Computing Implementation Management and SSategy", CRC Press.
- Michael Miller, "Cloud Computing: Web-Based Applications That Change the Way You Work and Collaborate Online".
- James E Smith, Ravi Nair, "Virtual Machines", Morgan Kaufmann Publishers.

ST. ALOYSIUS COLLEGE(AUTONOMOUS), JABALPUR

Reaccredited 'A++ 'Grade by NAAC(CGPA:3.58/4.00) College with Potential for Excellence by UGC

DST-FIST Supported & STAR College Scheme by DBT

List of Practical

1. Download and Install Virtual Machine (Virtual Box, VMware, and KVM)
2. Installing Virtual Machine
3. Controlling Virtual Machine (Start, restart, power off)
4. Editing Virtual Machine Hardware
5. Creating and Using Image Snapshots
6. Importing and Exporting Virtual Machine Images
7. Accessing Linux Command Line
8. Managing Files from the Command Line
9. Creating, Viewing, and Editing Text Files
10. Installing and Updating Software Packages

.

Faculty of Science

Bachelor of Science (B.Sc.) VI Semester

Paper-DSE-IIA

Web Technology using .NET

Course Outcomes

CO. No.	Course Outcomes	Cognitive Level
CO 1	Understand the basics of the Internet, the World Wide Web (WWW), and Client-server Computing.	U, A
CO 2	Have Knowledge of various web browsers, familiarized with Java scripting, Client-side scripting language, Web server Architecture, Database Connectivity (DBC)	K
CO 3	Have knowledge of HTML, its essential tags, Attributes, Text styles, Links to External Documents and different sections of an HTMLpage.	U
CO 4	Develop skills to generate HTML and haeknowledge of JavaScript and style sheets	U, An
CO 5	Have knowledge of Objects, Methods, Events and Functions and various types of text, styles	U

Credit and Marking Scheme

	Credits	Marks		Total Marks
		Internal	External	
Theory	3	40	60	**100**
Practical	1	40	60	**100**
Total	**4**	**200**		

Evaluation Scheme

	Marks	
	Internal	**External**
Theory	3 Internal Exams of 20 Marks (During the Semester) (Best 2 will be taken)	1 External Exams (At the End of Semester)
Practical	3 Internal Exams (During the Semester) (Best 2 will be taken)	1 External Exams (At the End of Semester)

No. of Lectures (in hours per week): 2 Hrs. per week

Total No. of Lectures: 60 Hrs. **Maximum Marks:** 60

Units	Topics	No. of Lecture
I	Topics Basics of Internet and Web: The basics of the Internet, World Wide Web, Web page, Home Page, Web site, Static, Dynamic and Active web page, Client server computing concepts, Web Browser, Client-Side Scripting, Server-Side Scripting, Introduction to HTML, Tags and Attributes.	10
II	Introduction to Style Sheet- Types, Selector, properties. Introduction to JavaScript- variable, operators, function, events, Array, Strings, Dialog Boxes.	10
III	Introduction to .NET- .NET Framework, .NET Architecture, CLR, the Just-in-Time Compiler, Garbage collection, .NET Framework class library.	10
IV	Introduction to ASP.NET- ASP.NET Page Life Cycle, Coding Model, Web forms, Web form controls, server controls, client controls, web forms, coding Models, Controls: Textbox, Label, Hyperlink, Button, Dropdown List, List Box, Check Box, Radio Button, File Upload, Validators, Master page.	10
V	ASP.NET Navigation Controls: Sitemap Path, Menu Control, Tree View Working with Database- Architecture of ADO.NET, Connected and Disconnected Database. Connection Class, Command Class, Data Adapter Class, and Dataset Class. Insert, Update, delete commands, and access the data from the database. Data Controls: From View, Grid View, etc.	20

References

Text Books:

- Web Technologies — Black Book — DreamTech Press
- Beginning HTML, XHTML, CSS and Javascript by John Duckett

Reference Books:

- HTML, XHTML and CSS Bible, 5th edition, Willey India-Steven M. Schafer
- Java EE and HTML-5 Enterprise Application Development (Oracle Press) by John Brock, Arun Gupta, Geertjan Wielenga.

Web Links:

- Internet technology course by NPTEL< nptel.ac.in>courses,
- www.udemy.com

List of Practical

1. Create a web form for addition of two numbers.
2. Create a web form for Simple Interest.
3. Create a web form for Factorial.
4. Create a web form for Prime number.
5. Create a web form for matching the value of two textboxes.
6. Create a web form for Calculator.
7. Create a web form for to demonstrate the session.
8. Create a web form with one list box and three check boxes named php, java, c respectively. On check and uncheck name of the check box should be added and removed to and from the list box.
9. Create a web form with one Drop Down List and demonstrate addition of items at first and last position. Show deletion also.
10. Demonstrate File Upload control.
11. Demonstrate Validation Controls.
12. Insert user data to Database through web form.
13. Create a sample college website and use Masterpage and Menu control.
14. Create Student Registration Form and corresponding database. Fetch the data into Grid View Control.

Faculty of Science

Bachelor of Science (B.Sc.) VI Semester

Paper-DSE-IIB

Text Mining using NLP

Course Outcomes

CO. No.	Course Outcomes	Cognitive Level
CO 1	Analyze and apply morphological analysis techniques such as lemmatization, finite automata, and finite state transducers.	U, A
CO 2	Perform Part-of-Speech (POS) tagging using rule-based and stochastic methods, and understand sequence labeling with HMM and Maximum Entropy models.	K
CO 3	Understand lexical semantics and perform word sense disambiguation using various approaches including dictionary-based methods and WordNet.	U
CO 4	Apply selectional restrictions and word similarity techniques using thesaurus and distributional methods for improved pragmatics and word sense disambiguation.	U, An
CO 5	Conduct discourse analysis, including anaphora and coreference resolution, and utilize lexical resources such as Penn Treebank, WordNet, and FrameNet.	U

Credit and Marking Scheme

	Credits	Marks		Total Marks
		Internal	External	
Theory	3	40	60	**100**
Practical	1	40	60	**100**
Total	**4**	**200**		

Evaluation Scheme

	Marks	
	Internal	**External**
Theory	3 Internal Exams of 20 Marks (During the Semester) (Best 2 will be taken)	1 External Exams (At the End of the Semester)
Practical	3 Internal Exams (During the Semester) (Best 2 will be taken)	1 External Exams (At the End of the Semester)

No. of Lectures (in hours per week): 2 Hrs. per week

Total No. of Lectures: 60 Hrs. **Maximum Marks:** 60

Units	Topics	No. of Lecture
I	History of NLP, Generic NLP system, levels of NLP, Knowledge in language processing, Ambiguity in Natural language, stages in NLP, challenges of NLP, Applications of NLP.	10
II	Morphology analysis –survey of English Morphology, Inflectional morphology & Derivational morphology, Lemmatization, Regular expression, finite automata, finite state transducers (FST), Morphological parsing with FST, Lexicon free FST Porter stemmer. N –Grams- N-gram language model, Self-learning topics: N-gram for spelling correction.	15
III	Part-Of-Speech tagging (POS)- Tag set for English (Penn Treebank), Rule-based POS tagging, Stochastic POS tagging, Issues –Multiple tags & words, Unknown words. Introduction to CFG, Sequence labeling: Hidden Markov Model (HMM), Maximum Entropy	10
IV	PRAGMATICS Selectional restrictions – Word Sense Disambiguation, WSD using Supervised, Dictionary & Thesaurus, Bootstrapping methods – Word Similarity using Thesaurus and Distributional methods.	10
V	Text summarization- LEXRANK, Optimization-based approaches for summarization, Summarization evaluation, Text classification. Sentiment Analysis introduction, Sentiment Analysis - Affective lexicons, Learning affective lexicons, Computing with affective lexicons, Aspect-based sentiment analysis.	15

TEXTBOOKS:

- Daniel Jurafsky, James H. Martin, "Speech and Language Processing: An Introduction to NaturalLanguageProcessing, Computational Linguistics and Speech", Pearson Publication, 2014.
- Steven Bird, Ewan Klein and Edward Loper, "Natural Language Processing with Python, First Edition,O'Reilly Media, 2009.

REFERENCE BOOK:

- Breck Baldwin, "Language Processing with Java and LingPipe Cookbook", Atlantic Publisher, 2015.
- Richard M Reese, "Natural Language Processing with Java", O'Reilly Media, 2015.

List of Practical

1. Design and implement an NLP pipeline that performs tokenization, lemmatization, POS tagging, and named entity recognition on a given text corpus.
2. Develop a morphological parser using finite state transducers (FST) for English words, and demonstrate its ability to handle inflectional and derivational morphology.
3. Construct an N-gram language model for a given text corpus and use it to perform tasks such as next-word prediction and spelling correction.
4. Implement rule-based and stochastic POS tagging on a sample text, and evaluate the accuracy of each method using the Penn Treebank tag set.
5. Train a Hidden Markov Model (HMM) for POS tagging and use it to tag a new text. Compare its performance with a Maximum Entropy model.
6. Implement a word sense disambiguation system using dictionary-based and supervised learning approaches. Evaluate the system on a set of ambiguous sentences.
7. Use WordNet to explore relationships among lexemes (homonymy, polysemy, synonymy, hyponymy) and implement a robust word sense disambiguation algorithm.
8. Perform discourse segmentation and anaphora resolution using Hobbs and Centering algorithms on a given text. Analyze the coherence and reference phenomena in the discourse.
9. Implement a text summarization system using LEXRANK or an optimization-based approach. Evaluate the summarization quality using standard evaluation metrics.
10. Develop an aspect-based sentiment analysis system that uses affective lexicons to analyze customer reviews. Implement the system and evaluate its accuracy on a given dataset.