LESSON PLAN

PROGRAM NAME: Computer Science (Honours)

SEM: 1st :: **PAPER: DC1**

COURSE: DISCRETE MATHEMATICS

TEACHER NAME: Deb Pratim Sinha

DC1: Discrete Mathematics: Number Systems: Introduction: Weighted and Non-Weighted Codes, positional, Binary, Octal, Hexadecimal, Binary coded Decimal (BCD), Gray Codes, Alphanumeric codes, ASCII, EBCDIC, Conversion of bases. Sets: finite and Infinite sets, uncountably Infinite Set; functions, relations, Properties of Binary Relations, Closure, Partial Ordering Relations; counting - Pigeonhole Principle, Permutation and Combination; Mathematical Induction, Principle of Inclusion and Exclusion. Recurrences: Recurrence Relations, generating functions, Linear Recurrence Relations with constant coefficients and their solution, Substitution Method, Master Theorem, Growth of Functions: Asymptotic Notations. Graph Theory :Basic Terminology, Models and Types, multi-graphs and weighted graphs, Graph Representation, Graph Isomorphism, Connectivity, Euler and Hamiltonian Paths and Circuits, Trees, Basic Terminology and properties of Trees, Binary tree, Introduction to Spanning Tree. Prepositional Logic Logical Connectives, Well-formed Formulas, Tautologies, Equivalences, Inference Theory.

Class	Topic
Lecture 1	Number Systems: Introduction: Weighted and Non-Weighted Codes.
Lecture 2	Number Systems: Positional, Binary, Octal, Hexadecimal, Binary coded Decimal (BCD), Gray Codes.
Lecture 3	Number Systems: Alphanumeric codes, ASCII, EBCDIC, Conversion of bases
Lecture 4	Sets: finite and Infinite sets, uncountably Infinite Set.
Lecture 5	Revision on Lecturer 1 to Lecturer 4.
Lecture 6	Revision on Lecturer 1 to Lecturer 4.
Lecture 7	**Class Test 1** on Lecturer 1 to Lecturer 4.
Lecture 8	Review on **Class Test 1.**
Lecture 9	Revision on **Class Test 1.**
Lecture 10	Revise Lecturer for weaker students on **Class Test 1.**
Lecture 11	Functions, relations, Properties of Binary Relations.
Lecture 12	Closure, Partial Ordering Relations.
Lecture 13	Counting - Pigeonhole Principle, Permutation and Combination.
Lecture 14	Mathematical Induction, Principle of Inclusion and Exclusion.
Lecture 15	Revision on Lecturer 11 to Lecturer 14.
Lecture 16	Revision on Lecturer 11 to Lecturer 14.
Lecture 17	**Class Test 2** on Lecturer 11 to Lecturer 14.
Lecture 18	Review on **Class Test 2.**

Lecture 19	Revise Lecturer for weaker students on **Class Test 2.**
Lecture 20	Recurrences: Recurrence Relations, generating functions.
Lecture 21	Linear Recurrence Relations with constant coefficients and their solution,
Lecture 23	Substitution Method, Master Theorem.
Lecture 24	Growth of Functions: Asymptotic Notations.
Lecture 25	Graph Theory :Basic Terminology, Models and Types
Lecture 26	Revision on Lecturer 11 to Lecturer 14.
Lecture 27	Revision on Lecturer 11 to Lecturer 14.
Lecture 27	Revision on Lecturer 20 to Lecturer 25.
Lecture 28	Revision on Lecturer 20 to Lecturer 25.
Lecture 29	**Class Test 3** on Lecturer 11 to Lecturer 14 and Lecturer 20 to Lecturer 25.
Lecture 30	Review on **Class Test 3.**
Lecture 31	Revise Lecturer for weaker students on **Class Test 3.**
Lecture 32	Revise Lecturer for weaker students on **Class Test 3.**
Lecture 33	Multi-graphs and weighted graphs, Graph Representation.
Lecture 34	Connectivity, Euler and Hamiltonian Paths and Circuits.
Lecture 35	Graph Isomorphism
Lecture 36	Trees, Basic Terminology and properties of Trees, Binary tree.
Lecture 37	Revision on Lecturer 20 to Lecturer 25.
Lecture 38	Revision on Lecturer 20 to Lecturer 25.
Lecture 39	Revision on Lecturer 33 to Lecturer 36.
Lecture 40	Revision on Lecturer 33 to Lecturer 36.
Lecture 41	**Class Test 4** on Revision on Lecturer 20 to Lecturer 25 and Lecturer 33 to Lecturer 36.
Lecture 42	Review on **Class Test 4.**
Lecture 43	Revise Lecturer for weaker students on **Class Test 4.**
Lecture 44	Revise Lecturer for weaker students on **Class Test 4.**
Lecture 45	Introduction to Spanning Tree. Prepositional Logic Logical Connectives.
Lecture 46	Well-formed Formulas, Tautologies.
Lecture 47	Equivalences, Inference Theory.
Lecture 48	Revision on Lecturer 33 to Lecturer 36.
Lecture 49	Revision on Lecturer 45 to Lecturer 47.
Lecture 50	**Class Test 5** on Lecturer 33 to Lecturer 36 and Revision on Lecturer 45 to Lecturer 47.
Lecture 51	Review on **Class Test 5.**
Lecture 52	Revise Lecturer for weaker students on **Class Test 5.**
Lecture 53	Revise Lecturer for weaker students on **Class Test 5.**
Lecture 54	Revise Lecturer on whole syllabus.

Lecture 55	Revise Lecturer on whole syllabus.
Lecture 56	Revise Lecturer on whole syllabus.
Lecture 57	**Class Test 6** on whole syllabus.
Lecture 58	Review on **Class Test 6.**
Lecture 59	Revise Lecturer for weaker students on **Class Test 6.**
Lecture 60	Revise Lecturer for weaker students on **Class Test 6.**
After completion of all lecturers Students will present a Seminar on this Course	
Course Outcome:	**After successfully completing this course, students will be able to:** Work with graphs and identify certain parameters. Develop the skill of converting mathematical problem graphically and vice versa. Motivates to solve real life problems. Develop suitable techniques of analysis of problems. Enable students to develop a positive attitude towards mathematics as an interesting and valuable subject to study. Develop the logical thinking of students. Apply mathematical foundations to design computer based algorithms. Perform certain algorithms, justify why these algorithms work, and give some estimates of the running times of these algorithms.

Text/Reference Books:

1. Elements of Discrete mathematics, C.L. Liu , D.P. Mahopatra, Tata McGraw Hill
2. Discrete Mathematics and Its Applications, Kenneth Rosen, McGraw Hill
3. Introduction to algorithms ,T.H. Coremen, C.E. Leiserson, R. L. Rivest, Prentice Hall on India,
4. Discrete Mathematics with Algorithms, M. O. Albertson and J. P. Hutchinson, John wiley Publication,
5. Discrete Structures, Logic, and Computability, J. L. Hein, , Jones and Bartlett Publishers,
6. Essentials of Discrete Mathematics, D.J. Hunter, Jones and Bartlett Publishers
7. Discrete Mathematical Structures with Applications to Combinatorics, V Ramaswamy, University Press
8. Discrete Mathematics: A Concept-based Approach, Basavaraj S Anami, Venkanna S Madalli, University Press

D. Sinha

Head
Dept. of Computer science & BCA
Gour Mahavidyalaya
Malda

GOUR MAHAVIDYALAYA
MANGALBARI, MALDA

LESSON PLAN

PROGRAM NAME: Computer Science (Honours)

SEM: 1st :: **PAPER:** DC2

COURSE: INTRODUCTION TO C PROGRAMMING

TEACHER NAME: Ekram Alam, Akhil Kr. Das & Arijit Bhattacharya

DC2: (a) Introduction to C Programming: **Introduction:** Basic Structure, Algorithms, Flowcharts, Structured programming constructs. **C Programming elements:** Character sets, Keywords, Constants, Variables, Data Types, Operators-Arithmetic, Relational, Logical and Assignment; Increment and Decrement and Conditional Operator, Precedence and Associations; Expressions, type casting. Comments, Functions, Storage Classes, Bit manipulation, Input and output. **C Pre-processor:** File inclusion, Macro substitution. **Statements:** Assignment, Control statements- if, if else, switch, break, continue, goto, Loops-while, do_while, for. **Functions:** Argument passing, return statement, return values and their types, recursion **Arrays:** String handling with arrays, String handling functions. **Pointers:** Definition and initialization, Pointer arithmetic, Pointers and arrays, String functions and manipulation, Dynamic storage allocation. **User defined Data types:** Structures. Structure arrays, Pointers to Functions and Structures, Unions, **File Access:** Opening, Closing, I/O operations.

CLASS	TOPIC
Lecturer 1	**Introduction:** Basic Structure, Algorithms.
Lecturer 2	**Introduction:** Flowcharts, Structured programming constructs.
Lecturer 3	**C Programming elements:** Character sets, Keywords, Constants, Variables, Data Types.
Lecturer 4	**C Programming elements:** Operators-Arithmetic, Relational, Logical and Assignment; Increment and Decrement and Conditional Operator,
Lecturer 5	**C Programming elements:** Precedence and Associations; Expressions, type casting.
Lecturer 6	**Class Test 1** on Lecturer 1 to Lecturer 5.
Lecturer 7	Review on **Class Test 1.**
Lecturer 8	Revision on **Class Test 1.**
Lecturer 9	Revise Lecturer for all students on **Class Test 1.**
Lecturer 10	**C Programming elements:** Comments, Functions, Storage Classes, Bit manipulation, Input and output.
Lecturer 11	**C Pre-processor:** File inclusion, Macro substitution.
Lecturer 12	**Statements:** Assignment, Control statements- if, if else, switch.
Lecturer 13	**Statements:** break, continue, goto, Loops-while, do_while, for.
Lecturer 14	Revision Lecturer on 1 to Lecturer 5 and Lecturer on 10 to Lecturer 13.
Lecturer 15	Revision Lecturer on 1 to Lecturer 5 and Lecturer on 10 to Lecturer 13.
Lecturer 16	Revision Lecturer on 1 to Lecturer 5 and Lecturer on 10 to Lecturer 13.
Lecturer 17	**Class Test 2** on Lecturer 1 to Lecturer 5 and Lecturer on 10 to Lecturer 13.
Lecturer 18	Revise Lecturer for all students on **Class Test 2.**
Lecturer 19	Revise Lecturer for all students on **Class Test 2.**
Lecturer 20	Revise Lecturer for all students on **Class Test 2.**
Lecturer 21	**Functions:** Argument passing, return statement, return values and their types, recursion.
Lecturer 22	Revision Lecturer on Lecturer 21.
Lecturer 23	Revision Lecturer on Lecturer 21.
Lecturer 24	**Arrays:** String handling with arrays, String handling functions.
Lecturer 25	Revision Lecturer on Lecturer 24.
Lecturer 26	Revision Lecturer on Lecturer 24.
Lecturer 27	Final revision Lecturer on Lecturer 21 and Lecturer 24.
Lecturer 28	**Class Test 3** on Lecturer on 10 to Lecturer 13 and Lecturer on 21 to Lecturer 24.
Lecturer 29	Revise Lecturer for all students on **Class Test 3.**
Lecturer 30	Revise Lecturer for all students on **Class Test 3.**
Lecturer 31	Revise Lecturer for all students on **Class Test 3.**
Lecturer 32	**Pointers:** Definition and initialization, Pointer arithmetic, Pointers and arrays.
Lecturer 33	Revision Lecturer on Lecturer 32.

Lecturer 34	Revision Lecturer on Lecturer 32.
Lecturer 35	**Pointers:** String functions and manipulation, Dynamic storage allocation.
Lecturer 36	Revision Lecturer on Lecturer 32 and Lecturer 35.
Lecturer 37	Revision Lecturer on Lecturer 32 and Lecturer 35.
Lecturer 38	**Class Test 4** on Lecturer on 21 to Lecturer 24 and Lecturer 32 and Lecturer 35.
Lecturer 39	Revise Lecturer for all students on **Class Test 4.**
Lecturer 40	Revise Lecturer for all students on **Class Test 4.**
Lecturer 41	Revise Lecturer for all students on **Class Test 4.**
Lecturer 42	**User defined Data types:** Structures. Structure arrays, Pointers to Functions and Structures, Unions
Lecturer 43	Revision Lecturer on Lecturer 42.
Lecturer 44	**File Access:** Opening, Closing, I/O operations.
Lecturer 45	Revision Lecturer on Lecturer 44.
Lecturer 46	Revise Lecturer for all students on whole syllabus.
Lecturer 47	Revise Lecturer for all students on whole syllabus.
Lecturer 48	Revise Lecturer for all students on whole syllabus.
Lecturer 49	**Class Test 5** on Whole syllabus.
Lecturer 50	Review on **Class Test 5.**
Lecturer 51	Revision on **Class Test 5.**
Lecturer 52	Revise Lecturer for all students on **Class Test 5.**
Lecturer 53	Revise Lecturer for all students on class test 5.
Lecturer 54	Revise Lecturer for all students on class test 5.
Lecturer 55	Revise Lecturer for all students on class test 5.
Lecturer 56	Revise Lecturer on whole Syllabus.
Lecturer 57	Revise Lecturer on whole Syllabus.
Lecturer 58	Revise Lecturer on whole Syllabus.
Lecturer 59	Revise Lecturer on whole Syllabus.
Lecturer 60	Revise Lecturer on whole Syllabus.
After completion of all lecturers Students will present a Seminar on this Course	
Course Outcome:	**After successfully completing this course, students will be able to:** Understand algorithms and flowchart for solving problems using computers. Understand and can choose the loops and decision-making statements to solve the problem. Implement different Operations on arrays and will use functions to solve the given problem. Implement different Operations using pointers, ADT and file system to solve the given problem. To enrich the students in logic development required for programming. To help the students to build carrier in various branches of software development.

Text/Reference Books:

1. Programming with C, Byron S. Gottfried, McGraw Hill.
2. The C Programming Language, Kernighan and Dennis, PHI.
3. The Complete reference C, Herbert Schildt, McGraw Hill.
4. Let Us C, Kanitkar, BPB Publication.
5. Programming in ANSI C, Balaguruswamy, McGraw Hill.
6. Programming Languages, Allen B. Tucker, Tata McGraw Hill.

GOUR MAHAVIDYALAYA MANGALBARI, MALDA

Head
Dept. of Computer science & BCA
Gour Mahavidyalaya
Malda

LESSON PLAN

PROGRAM NAME: Computer Science (Honours)

SEM: 1st :: **PAPER:** DC2 (b)

COURSE: INTRODUCTION TO C PROGRAMMING

TEACHER NAME: Ekram Alam, Akhil Kr. Das & Arijit Bhattacharya

DC2: (b) C Programming Lab:	
CLASS	**TOPIC**
Lecturer 1	1. Write A Program to print the sum and product of digits of an integer. 2. Write A Program to reverse a number.
Lecturer 2	3. Write A Program to compute the sum of the first n terms of the following series, S =1+1/2 +1/3+1/4+..... 4. WAP to compute the sum of the first n terms of the following series, S =1-2+3-4+5 ....
Lecturer 3	5. Write a function that checks whether a given string is Palindrome or not. Use this function to find whether the string entered by user is Palindrome or not. 6. Write a function to find whether a given no. is prime or not. Use the same to generate the prime numbers less than 100.
Lecturer 4	7. Write A Program to compute the factors of a given number. 8. Write A Program a macro that swaps two numbers. Write A Program to use it.
Lecturer 5	9. Write A Program P to print a triangle of stars as follows (take number of lines from user): * *** ***** ******* *********
Lecturer 6	Revision Lecture on Lecture 1 to Lecture 5
Lecturer 7	Class Test 1 on Lecturer 1 to Lecturer 5.
Lecturer 8	Review on Class Test 1.
Lecturer 9	Revision on Class Test 1.
Lecturer 10	10. Write A Program to perform following actions on an array entered by the user : i) Print the even-valued elements. ii) Print the odd-valued elements. iii) Calculate and print the sum and average of the elements of array.
Lecturer 11	10. Write A Program to perform following actions on an array entered by the user : iv) Print the maximum and minimum element of array v) Remove the duplicates from the array vi) Print the array in reverse order. The program should present a menu to the user and ask for one of the options. The menu should also include options to re-enter array and to quit the program.
Lecturer 12	11. WAP that prints a table indicating the number of occurrences of each alphabet in the text entered as command line arguments. 12. Write a program that swaps two numbers using pointers.
Lecturer 13	Class Test 2 on Lecturer 10 to Lecturer 12.
Lecturer 14	Review on Class Test 2.
Lecturer 15	Revision on Class Test 2.
Lecturer 16	13. Write a program in which a function is passed address of two variables and then alter its contents. 14. Write a program which takes the radius of a circle as input from the user, passes it to another function that computes the area and the circumference of the circle and displays the value of area and circumference from the main() function.
Lecturer 17	15. Write a program to find sum of n elements entered by the user. To write this program, allocate memory dynamically using malloc() / calloc() functions or new operator. 16. Write a menu driven program to perform following operations on strings:

	a) Show address of each character in string b) Concatenate two strings without using strcat function. c) Concatenate two strings using strcat function. d) Compare two strings e) Calculate length of the string (use pointers)
Lecturer 18	Revision Lecturer on Lecturer 10 to Lecturer 12.
Lecturer 19	Revision Lecturer on Lecture 16 to Lecture 17.
Lecturer 20	**Class Test 3** on Lecturer 10 to Lecturer 12 and Lecture 16 to Lecture 17.
Lecturer 21	Review on **Class Test 3.**
Lecturer 22	Revision on **Class Test 3.**
Lecturer 23	**16.** Write a menu driven program to perform following operations on strings: f) Convert all lowercase characters to uppercase g) Convert all uppercase characters to lowercase h) Calculate number of vowels i) Reverse the string **17.** Given two ordered arrays of integers, write a program to merge the two-arrays to get an ordered array.
Lecturer 24	**18.** Write a program to display Fibonacci series: **(i)** using recursion, (ii) using iteration. **19.** WAP to calculate Factorial of a number I) using recursion, (ii) using iteration.
Lecturer 25	Revision Lecturer on Lecture 16 to Lecture 17.
Lecturer 26	Revision Lecturer on Lecture 23 to Lecturer 24.
Lecturer 27	**Class Test 4** on Lecture 16 to Lecture 17 and Lecturer 23 to Lecturer 24.
Lecturer 28	Review on **Class Test 4.**
Lecturer 29	Revision on **Class Test 4.**
Lecturer 30	**20.** WAP to calculate GCD of two numbers (i) With recursion (ii) Without recursion.
Lecturer 31	**21.** Create Matrix class using templates. Write a menu-driven program to perform following Matrix operations (2-D array implementation): a) Sum b) Difference
Lecturer 32	**21.** Create Matrix class using templates. Write a menu-driven program to perform following Matrix operations (2-D array implementation): c) Product d) Transpose **22.** Copy the contents of one text file to another file, after removing all whitespaces.
Lecturer 33	**23.** Write a function that reverses the elements of an array in place. The function must accept only one pointer value and return void. **24.** Write a program that will read 10 integers from user and store them in an array.
Lecturer 34	**Class Test 5** on Lecturer 23 to Lecturer 24 and Lecturer 30 to Lecturer 33.
Lecturer 35	Review on **Class Test 5.**
Lecturer 36	Revision on **Class Test 5.**
Lecturer 37	Revise Lecturer for all students on whole syllabus.
Lecturer 38	Revise Lecturer for all students on whole syllabus.
Lecturer 39	Revise Lecturer for all students on whole syllabus.
Lecturer 40	Revise Lecturer for all students on whole syllabus.
After completion of all lecturers Students will present a Seminar on this Course	
Course Outcome:	**After successfully completing this course, students will be able to:** Understand algorithms and flowchart for solving problems using computers. Understand and can choose the loops and decision-making statements to solve the problem. Implement different Operations on arrays and will use functions to solve the given problem. Implement different Operations using pointers, ADT and file system to solve the given problem. To enrich the students in logic development required for programming. To help the students to build carrier in various branches of software development.

Text/Reference Books:

1. Programming with C, Byron S. Gottfried, McGraw Hill.
2. The C Programming Language, Kernighan and Dennis, PHI.
3. The Complete reference C, Herbert Schildt, McGraw Hill.
4. Let Us C, Kanitkar, BPB Publication.
5. Programming in ANSI C, Balaguruswamy, McGraw Hill.
6. Programming Languages, Allen B. Tucker, Tata McGraw Hill.

Head
Dept. of Computer science & BCA
Gour Mahavidyalaya
Malda

GOUR MAHAVIDYALAYA

LESSON PLAN

PROGRAM NAME: Computer Science (Honours)

SEM: 2nd :: **PAPER: DC3 (a)**

COURSE: DATA STRUCTURE & ALGORITHM

TEACHER NAME: Ekram Alam & Akhil Kr. Das

DC3: a) Data Structure & Algorithm: **Introduction to Data Structure:** Abstract Data Type. **Arrays:** 1D, 2D and Multi-dimensional Arrays, Sparse Matrices. Polynomial representation (Polynomial Representation as Application). **Linked Lists:** Singly, Doubly and Circular Lists; Polynomial representation (Polynomial Representation as Application). **Stacks:** Implementing single / multiple stacks in an Array; Prefix, Infix and Postfix expressions, Utility and conversion of these expressions from one to another; Applications of stack; Limitations of Array representation of stack. **Queues:** Array and Linked representation of Queue, Circular Queue, De-queue, Priority Queues. **Recursion:** Developing Recursive Definition of Simple Problems and their implementation; Advantages and Limitations of Recursion; Understanding what goes behind Recursion (Internal Stack Implementation). **Trees:** Introduction to Tree as a data structure; Binary Trees (Insertion, Deletion, Recursive and Iterative Traversals on Binary Search Trees; Height-Balanced Trees (Various operations on AVL Trees). **Searching and Sorting:** Linear Search, Binary Search, Comparison of Linear and Binary Search, Sort: Bubble sort, Selection Sort, Insertion Sort, Merge Sort, Quick sort, Heap Sort, Comparison of Sorting Techniques. **Hashing:** Introduction to Hashing, Choosing a Hash Function, collision resolution techniques.

CLASS	TOPIC
Lecturer 1	Introduction: Algorithms, ADT.
Lecturer 2	Arrays: One dimensional, Two Dimensional and Multi-dimensional Arrays.
Lecturer 3	Sparse Matrices. Polynomial representation (Polynomial Representation as Application).
Lecturer 4	**Linked Lists:** Singly, Doubly and Circular Lists.
Lecturer 5	Polynomial representation (Polynomial Representation as Application) and Revision on Lecture 1 to Lecture 4.
Lecturer 6	**Class Test 1** on Lecturer 1 to Lecturer 5.
Lecturer 7	Review on **Class Test 1.**
Lecturer 8	Revision on **Class Test 1.**
Lecturer 9	Revise Lecturer for all students on **Class Test 1.**
Lecturer 10	Revise Lecturer for all students on **Class Test 1.**
Lecturer 11	**Stacks:** Implementing single / multiple stacks in an Array, Prefix, Infix and Postfix expressions.
Lecturer 12	**Stacks:** Utility and conversion of above expressions from one to another; Applications of stack; Limitations of Array representation of stack.
Lecturer 13	**Queues:** Array and Linked representation of Queue.
Lecturer 14	**Queues:** Circular Queue, De-queue, Priority Queues.
Lecturer 15	**Class Test 2** on Lecturer 1 to Lecturer 4 and Lecturer 10 to Lecturer 13.
Lecturer 16	Review on **Class Test 2.**
Lecturer 17	Revision on **Class Test 2.**
Lecturer 18	Revise Lecturer for all students on **Class Test 2.**
Lecturer 19	Revise Lecturer for all students on **Class Test 2.**
Lecturer 20	**Recursion:** Developing Recursive Definition of Simple Problems and their implementation.
Lecturer 21	**Recursion:** Advantages and Limitations of Recursion; Understanding what goes behind Recursion (Internal Stack Implementation).
Lecturer 22	**Class Test 3** on Lecturer 10 to Lecturer 13 and Lecturer 20 to Lecturer 21.
Lecturer 23	Review on **Class Test 3.**
Lecturer 24	Revision on **Class Test 3.**

Lecturer 25	Revise Lecturer for all students on **Class Test 3.**
Lecturer 26	Revise Lecturer for all students on **Class Test 3.**
Lecturer 27	**Trees:** Introduction to Tree as a data structure; Binary Trees (Insertion, Deletion).
Lecturer 28	Recursive and Iterative Traversals on Binary Search Trees.
Lecturer 29	Revision Lecture on Lecture 28 and Height-Balanced Trees (Various operations on AVL Trees).
Lecturer 30	**Searching** Linear Search, Binary Search, Comparison of Linear and Binary Search,
Lecturer 31	Revision on lecture 27 to Lecture 30.
Lecturer 32	Revision on lecture 27 to Lecture 30.
Lecturer 33	**Class Test 4** on Lecturer 20 to Lecturer 21and Lecturer 27 to Lecturer 30.
Lecturer 34	Review on **Class Test 4.**
Lecturer 35	Revision on **Class Test 4.**
Lecturer 36	Revise Lecturer for all students on **Class Test 4.**
Lecturer 37	Revise Lecturer for all students on **Class Test 4.**
Lecturer 38	**Sorting:** Bubble sort, Selection Sort.
Lecturer 39	**Sorting:** Insertion Sort, Merge Sort,
Lecturer 40	**Sorting:** Quick sort, Heap Sort,
Lecturer 41	**Sorting:** Comparison among all Sorting Techniques.
Lecturer 42	Revision on Lecture 38 to Lecture 39.
Lecturer 43	Revision on Lecture 40 to Lecture 41.
Lecturer 44	**Class Test 5** on Lecturer 27 to Lecturer 30 and Lecture 38 to Lecture 41.
Lecturer 45	Review on **Class Test 5.**
Lecturer 46	Revision on **Class Test 5.**
Lecturer 47	Revise Lecturer for all students on **Class Test 5.**
Lecturer 48	Revise Lecturer for all students on **Class Test 5.**
Lecturer 49	**Hashing:** Introduction to Hashing, Choosing a Hash Function.
Lecturer 50	**Hashing:** Collision resolution techniques.
Lecturer 51	Revision on Lecture 49 to Lecture 50.
Lecturer 52	**Class Test 6** on whole syllabus
Lecturer 53	Review on **Class Test 6.**
Lecturer 54	Revision on **Class Test 6.**
Lecturer 55	Revise Lecturer for all students on **Class Test 6.**
Lecturer 56	Revise Lecturer for all students on **Class Test 6.**
Lecturer 57	Revision Lecturer on whole syllabus.
Lecturer 58	Revision Lecturer on whole syllabus.
Lecturer 59	Revision Lecturer on whole syllabus.
Lecturer 60	Revision Lecturer on whole syllabus.
After completion of all lecturers Students will present a Seminar on this Course	
Course Outcome:	**After successfully completing this course, students will be able to:** Students will understand system related Programming such as Operating System functioning. Students will capable to develop problem solving abilities using a computer. To build the necessary skill set and analytical abilities for developing computer based solutions for real life problems. To imbibe quality software development practices. To create awareness about process and product standards. Students will train in professional skills related to Software Industry. To prepare necessary knowledge which is related to operating system and base for research and development in Computer Science.

Text/Reference Books:

1. Fundamentals of Data Structures in C, Ellis Horowitz, Sartaj Sahni, Susan Anderson-Freed, Silicon Pr.
2. Data Structures: A Pseudocode Approach with C, Richard F. Gilberg and Behrouz A.Forouzan, Cengage Learning
3. Data Structures In C, Noel Kalicharan, CreateSpace Independent Publishing Platform.
4. Data Structures and algorithm in C, Adam Drozdek, Cengage Learning.

5] The C Programming Language, Brian W. Kernighan and Dennis Ritchie, PrenticeHall.

6] Data Structures, Algorithms and applications in C++, Sartaj Sahni,Universities Press.

7] Data Structures Using C and C++, Aaron M. Tanenbaum, Moshe J. Augenstein, Yedidyah Langsam, PHI.

8] Classic Data Structures, Debasis Samanta, PHI

9] Fundamental of Computer Algorithms, Horowitz, Sahni, Rajasekaran, Universities Press.

Head
Dept. of Computer science & BCA
Gour Mahavidyalaya
Malda

GOUR MAHAVIDYALAYA
MANGALBARI, MALDA

LESSON PLAN

PROGRAM NAME: Computer Science (Honours)

SEM: 2nd :: **PAPER: DC3 (b)**

COURSE: DATA STRUCTURE LAB

TEACHER NAME: Ekram Alam & Akhil Kr. Das

DC3: b) Data Structure Lab:	
CLASS	**TOPIC**
Lecturer 1	1. Write a program to search an element from a list. Give user the option to perform Linear or Binary search. 2. WAP using templates to sort a list of elements. Give user the option to perform sorting using Insertion sort, Bubble sort, Selection sort etc.
Lecturer 2	3. Implement Singly Linked List. Include functions for insertion, deletion and search of a number, reverse the list and concatenate two linked lists. 4. Implement Doubly Linked List using templates. Include functions for insertion, deletion and search of a number, reverse the list.
Lecturer 3	5. Implement Circular Linked List. Include functions for insertion, deletion and search of a number, reverse the list.
Lecturer 4	**Class Test 1** on Lecturer 1 to Lecturer 3.
Lecturer 5	Review on **Class Test 1.**
Lecturer 6	Revision on **Class Test 1.**
Lecturer 7	Revise Lecturer for all students on **Class Test 1.**
Lecturer 8	Revise Lecturer for all students on **Class Test 1.**
Lecturer 9	6. Perform Stack operations using Linked List implementation. 7. Perform Stack operations using Array implementation.
Lecturer 10	8. Perform Queues operations using Circular Array implementation. 9. Create and perform different operations on Double-ended Queues using Linked List implementation.
Lecturer 11	10. WAP to scan a polynomial using linked list and add two polynomial.
Lecturer 12	**Class Test 2** on Lecturer 1 to Lecturer 3 and Lecturer 9 to Lecturer 11.
Lecturer 13	Review on **Class Test 2.**
Lecturer 14	Revision on **Class Test 2.**
Lecturer 15	Revise Lecturer for all students on **Class Test 2.**
Lecturer 16	Revise Lecturer for all students on **Class Test 2.**
Lecturer 17	11. WAP to calculate factorial and to compute the factors of a given no. (i)using recursion, (ii) using iteration 12. WAP to display Fibonacci series (i)using recursion, (ii) using iteration.
Lecturer 18	13. Write A Program to calculate GCD of 2 number (i) With recursion (ii) Without recursion.
Lecturer 19	14. Write A Program to create a Binary Search Tree and include following operations in tree: (a) Insertion (Recursive and Iterative Implementation)
Lecturer 20	**Class Test 3** on Lecturer 9 to Lecturer 11 and Lecturer 17 to Lecturer 19.
Lecturer 21	Review on **Class Test 3.**
Lecturer 22	Revision on **Class Test 3.**
Lecturer 23	Revise Lecturer for all students on **Class Test 3.**
Lecturer 24	Revise Lecturer for all students on **Class Test 3.**
Lecturer 25	14. Write A Program to create a Binary Search Tree and include following operations in tree: (b) Deletion by copying (c) Deletion by Merging (d) Search a no. in BST
Lecturer 26	14. Write A Program to create a Binary Search Tree and include following operations in tree: (e) Display its preorder, postorder and inorder traversals Recursively

	(f) Display its preorder, postorder and inorder traversals Iteratively (g) Display its level-by-level traversals
Lecturer 27	14. Write A Program to create a Binary Search Tree and include following operations in tree: (h) Count the non-leaf nodes and leaf nodes (i) Display height of tree (j) Create a mirror image of tree (k) Check whether two BSTs are equal or not
Lecturer 28	15. WAP to reverse the order of the elements in the stack using additional stack.
Lecturer 29	Revision Lecture on whole syllabus.
Lecturer 30	Revision Lecture on whole syllabus.
Lecturer 31	Revision Lecture on whole syllabus.
Lecturer 32	**Class Test 4 on Whole syllabus.**
Lecturer 33	Review on **Class Test 4.**
Lecturer 34	Review on **Class Test 4.**
Lecturer 35	Revise Lecturer for all students on **Class Test 4.**
Lecturer 36	Revise Lecturer for all students on **Class Test 4.**
Lecturer 37	Revision Lecture on whole syllabus.
Lecturer 38	Revision Lecture on whole syllabus.
Lecturer 39	Revision Lecture on whole syllabus.
Lecturer 40	Revision Lecture on whole syllabus.

Text/Reference Books:

1. Fundamentals of Data Structures in C, Ellis Horowitz, Sartaj Sahni, Susan Anderson-Freed, Silicon Pr.
2. Data Structures: A Pseudocode Approach with C, Richard F. Gilberg and Behrouz A.Forouzan, Cengage Learning
3. Data Structures In C, Noel Kalicharan, CreateSpace Independent Publishing Platform.
4. Data Structures and algorithm in C, Adam Drozdek, Cengage Learning.
5. The C Programming Language, Brian W. Kernighan and Dennis Ritchie, PrenticeHall.
6. Data Structures, Algorithms and applications in C++, Sartaj Sahni,Universities Press.
7. Data Structures Using C and C++, Aaron M. Tanenbaum, Moshe J. Augenstein, Yedidyah Langsam, PHI.
8. Classic Data Structures, Debasis Samanta, PHI
9. Fundamental of Computer Algorithms, Horowitz, Sahni, Rajasekaran, Universities Press.

GOUR MAHAVIDYALAYA MANGALBARI, MALDA

Head
Dept. of Computer science & BCA
Gour Mahavidyalaya
Malda

LESSON PLAN

PROGRAM NAME: Computer Science (Honours)

SEM: 2nd :: **PAPER: DC–4a**

COURSE: DIGITAL LOGIC SYSTEM

TEACHER NAME: Arijit Bhattacharya

DC–4 : Digital Logic System

Boolean Algebra: Fundamentals of Boolean Expression: Definition of Switching Algebra, Basic properties of Switching Algebra, Huntington's Postulates, Basic logic gates (AND, OR, NOT), De-Morgan's Theorem, Universal Logic gates (NAND, NOR), Minterm, Maxterm, Minimization of Boolean Functions using K-Map, Simplification of logic expression.

Combinational Circuits: Half adders, Full Adder, Half Subtractor, Full Subtractor and construction using Basic Logic Gates (OR, AND, NOT) and Universal Logic Gates (NAND & NOR), Multibit Adder- Ripple Carry Adder, Carry Look Ahead adder, BCD Adder, Adder/Subtractor unit Construction using 4 bit Full adders units, 1 bit, 2 bit and 3 bit Comparators.

Data Selector-Multiplexer: Expansion (Cascading), Function Realization.

Encoders:- Realization of simple Encoders and priority Encoders using Basic and Universal Logic gates.

Data Distributor:- De-multiplexer, Cascading. Chip Selector/Minterm Generator - Decoder- Function Realization, Cascading, BCD Decoders, Seven Segment Display and Decoders, realization of seven segment decoders using basic gates. Parity bit and Code Converters: Parity bit Generator/Checker, Gray to Binary code converter, Binary to Gray Code Converter.

Sequential Circuits: Set/Reset (SR) Latch: Using NAND and NOR gates, Gated S-R latches, D Latch, J-K Latch, T Flip Flop, Race around Condition, Master Slave J-K Flip Flop, Edge Triggered SR, D and JK Flip Flop, Flip-Flop Conversions, Flip-Flops with Preset and Clear

Registers: Serial Input Serial Output, Serial Input Parallel Output, Parallel input Serial Output, Parallel Input parallel Output, Universal Shift Registers.

Counters: Asynchronous Counter: UP/DOWN Counters, Mod - N Counters, BCD Counter, Synchronous Counter: UP/DOWN Counters, Mod-N Counters, Ring Counters, Johnson Counters.

Class	Topic
Lecture 1	Fundamentals of Boolean Expression: Definition of Switching Algebra, Basic properties of Switching Algebra, Huntington's Postulates, Basic logic gates (AND, OR, NOT)
Lecture 2	De-Morgan's Theorem, Universal Logic gates (NAND, NOR)
Lecture 3	Minterm, Maxterm, Minimization of Boolean Functions using K-Map, Simplification of logic expression
Lecture 4	Simplification of logic expression
Lecture 5	Revision on Lecturer 1 to Lecturer 4.
Lecture 6	Class Test 1 on Lecturer 1 to Lecturer 4.
Lecture 7	Review on **Class Test 1.**
Lecture 8	Revision on **Class Test 1.**
Lecture 9	Revise Lecturer for weaker students on **Class Test 1.**
Lecture 10	Half adders, Full Adder
Lecture 11	Half Subtractor, Full Subtractor
Lecture 12	Half adders, Full Adder, Half Subtractor, Full Subtractor construction using Basic Logic Gates (OR, AND, NOT) and Universal Logic Gates (NAND & NOR)
Lecture 13	Multibit Adder- Ripple Carry Adder
Lecture 14	Revision on Lecturer 10 to Lecturer 13.

Lecture 15	**Class Test 2** on Lecturer 10 to Lecturer 13.
Lecture 16	Review on **Class Test 2.**
Lecture 17	Revise Lecturer for weaker students on **Class Test 2.**
Lecture 18	Carry Look Ahead adder, BCD Adder
Lecture 19	Adder/Subtractor unit Construction using 4 bit Full adders units
Lecture 20	Adder/Subtractor unit Construction using 4 bit Full adders units – continued
Lecture 21	1 bit, 2 bit and 3 bit Comparators
Lecture 23	Data Selector-Multiplexer: Expansion (Cascading), Function Realization.
Lecture 24	Encoders : Realization of simple Encoders and priority Encoders using Basic Logic gates
Lecture 25	Realization of simple Encoders and priority Encoders using Universal Logic gates
Lecture 26	Revision on Lecturer 10 to Lecturer 13.
Lecture 27	Revision on Lecturer 10 to Lecturer 13.
Lecture 27	Revision on Lecturer 18 to Lecturer 25.
Lecture 28	Revision on Lecturer 18 to Lecturer 25.
Lecture 29	**Class Test 3** on Lecturer 10 to Lecturer 13 and Lecturer 18 to Lecturer 25.
Lecture 30	Review on **Class Test 3.**
Lecture 31	Revise Lecturer for weaker students on **Class Test 3.**
Lecture 32	Revise Lecturer for weaker students on **Class Test 3.**
Lecture 33	Data Distributor :- De-multiplexer, Cascading. Chip Selector/Minterm Generator
Lecture 34	Decoder - Function Realization, Cascading, BCD Decoders
Lecture 35	Seven Segment Display and Decoders, realization of seven segment decoders using basic gates
Lecture 36	Parity bit and Code Converters: Parity bit Generator/Checker
Lecture 37	Gray to Binary code converter, Binary to Gray Code Converter
Lecture 38	Revision on Lecturer 18 to Lecturer 25.
Lecture 39	Revision on Lecturer 18 to Lecturer 25.
Lecture 40	Revision on Lecturer 33 to Lecturer 37.
Lecture 41	Revision on Lecturer 33 to Lecturer 37.
Lecture 42	**Class Test 4** on Lecturer 18 to Lecturer 25 and Lecturer 33 to Lecturer 37.
Lecture 43	Review on **Class Test 4.**
Lecture 44	Revise Lecturer for weaker students on **Class Test 4.**
Lecture 45	Revise Lecturer for weaker students on **Class Test 4.**
Lecture 46	Set/Reset (SR) Latch: Using NAND and NOR gates, Gated S-R latches, D Latch,
Lecture 47	J-K Latch, T Flip Flop, Race around Condition
Lecture 48	J Master Slave J-K Flip Flop, Edge Triggered SR
Lecture 49	D and JK Flip Flop, Flip-Flop Conversions, Flip-Flops with Preset and Clear
Lecture 50	Revision on Lecturer 33 to Lecturer 37.

Lecture 51	Revision on Lecturer 33 to Lecturer 37.
Lecture 52	Revision on Lecturer 46 to Lecturer 50.
Lecture 53	Class Test 5 on Lecturer 33 to Lecturer 37 and Lecturer 46 to Lecturer 50.
Lecture 54	Review on Class Test 5.
Lecture 55	Registers: Serial Input Serial Output, Serial Input Parallel Output, Parallel input Serial Output, Parallel Input parallel Output, Universal Shift Registers.
Lecture 56	Asynchronous Counter: UP/DOWN Counters, Mod - N Counters, BCD Counter
Lecture 57	Synchronous Counter: UP/DOWN Counters, Mod-N Counters, Ring Counters, Johnson Counters
Lecture 58	Class Test 6 on Whole syllabus
Lecture 59	Review on Class Test 6.
Lecture 60	Review of Class Test 6 and special Lecturer for weaker students.
After completion of all lecturers Students will present a Seminar on this Course	
Course Outcome:	**After completion of this course student will be able to:** Solve problems based on inter conversion of number systems. Reduce the expression using Boolean theorems. Reduce expressions using K maps in SOP and POS forms. Understand the operation of all types of Logic Gates, their families etc. Understand how to use Combinational Logic circuits using Logic Gates and using ICs.

Text/Reference Books:

1. Digital Circuits, Vol - I & II, D. Ray Chaudhuri, Platinum Publishers.
2. Digital Systems - Principle & Applications, Tocci & Widmer, EEE.
3. Digital Logic & State Machine Design, Comer, Oxford.
4. Digital Principle & Applications, Malvino & Leach, McGraw Hill.
5. Digital Design, Mano, PHI.
6. Digital Integrated Electronics- H.Taub & D.Shilling, Mc Graw Hill.
7. Digital Circuits and Design, Salivahan, Vikas.

GOUR MAHAVIDYALAYA
MANGALBARI, MALDA

Head
Dept. of Computer science & BCA
Gour Mahavidyalaya
Malda

LESSON PLAN

PROGRAM NAME: Computer Science (Honours)

SEM: 2nd :: **PAPER:** DC4 (b)

COURSE: DIGITAL LOGIC LAB

TEACHER NAME: Arijit Bhattacharya

DC4: b) Digital Logic Lab:	
CLASS	**TOPIC**
Lecturer 1	1. Implementation of different functions using Basic and Logic gates, SOP, POS.
Lecturer 2	2. Study and prove De-Morgan's Theorem.
Lecturer 3	3. Universal function using NAND and NOR gates.
Lecturer 4	4. Implementation of half and Full adder (3-bit) using basic logic gates.
Lecturer 5	4. Implementation of half and Full adder (3-bit) using Universal logic gates (NAND & NOR).
Lecturer 6	**Class Test 1** on Lecturer 1 to Lecturer 5.
Lecturer 7	Review on **Class Test 1.**
Lecturer 8	Revision on **Class Test 1.**
Lecturer 9	5. Implementation of half and Full Subtractor (3-bit) using basic logic gates and Universal logic gates (NAND & NOR).
Lecturer 10	6. 1 Digit BCD adder using 7483 and other logic gates.
Lecturer 11	7. Design 4 to 1 multiplexer using logic/Universal gates
Lecturer 12	7. Design 4 to 1 multiplexer implement full adder/full subtractor.
Lecturer 13	**Class Test 2** on Lecturer 9 to Lecturer 12.
Lecturer 14	Review on **Class Test 2.**
Lecturer 15	Revision on **Class Test 2.**
Lecturer 16	8. Using 74153 and 74151 to implement full adder/ full subtractor and other functions.
Lecturer 17	9. Cascading of Multiplexers.
Lecturer 18	10. Design 2 to 4 decoder using basic / universal logic gates.
Lecturer 19	11. Study 74138 and 74139 and implement full adder / full subtractor and other functions.
Lecturer 20	**Class Test 3** on Lecturer 16 to Lecturer 19.
Lecturer 21	Review on **Class Test 3.**
Lecturer 22	Revision on **Class Test 3.**
Lecturer 23	12. Implementation of 1 bit Comparator using decoders.
Lecturer 24	13. Cascading of Decoders. 14. Design a parity generator and checker using basic gates.
Lecturer 25	15. Construct and study comparators using 7485.
Lecturer 26	16. Construct Comparator (2-bit) using logic gates.
Lecturer 27	**Class Test 4** on Lecturer 16 to Lecturer 19.
Lecturer 28	Review on **Class Test 4.**
Lecturer 29	Revision on **Class Test 4.**
Lecturer 30	17. Design a seven segment display unit using Common anode/Common cathode and 7447 / 7448. 18. Study Priority Encoder Chip 74147/74148.
Lecturer 31	19. Realization of RS, D, JK Clocked/Gated Level Triggered Flip-Flop using basic/Universal logic gates.
Lecturer 32	20. Study and Conversion of Flip-Flops: D to JK, JK to D, JK to T.
Lecturer 33	20. Study and Conversion of Flip-Flops: SR to JK, SR to D Flip-flop.
Lecturer 34	21. Design synchronous and asynchronous counters MOD-n (MOD-8, MOD-10) UP/ DOWN and connecting Seven Segment Display along with decoder for display of counting sequence.
Lecturer 35	22. Construction of ODD/EVEN 4 bit Synchronous Counter.
Lecturer 36	23. 2-bit Universal Shift Register.
Lecturer 37	Revise Lecturer for all students on whole syllabus for their own choice.
Lecturer 38	Revise Lecturer for all students on whole syllabus for their own choice.
Lecturer 39	Revise Lecturer for all students on whole syllabus for their own choice.
Lecturer 40	Revise Lecturer for all students on whole syllabus for their own choice.

Text/Reference Books:

1. Programming with C, Byron S. Gottfried, McGraw Hill.
2. The C Programming Language, Kernighan and Dennis, PHI.
3. The Complete reference C, Herbert Schildt, McGraw Hill.
4. Let Us C, Kanitkar, BPB Publication.
5. Programming in ANSI C, Balaguruswamy, McGraw Hill.
6. Programming Languages, Allen B. Tucker, Tata McGraw Hill.

Head
Dept. of Computer science & BCA
Gour Mahavidyalaya
Malda

GOUR MAHAVIDYALAYA
MANGALBARI MALDA

LESSON PLAN

PROGRAM NAME: Computer Science (Honours)

SEM: 3rd :: **PAPER: DC 5**

COURSE: COMPUTER ORGANIZATION & ARCHITECTURE

TEACHER NAME: Dr. Subhendu Chatterjee & Shib Charan Chowdhury

DC5: Computer Organization & Architecture: Basic Structure of Computers: Basic Functional Units, Basic Operational Concept, Bus Structure, Software, Performance, Multiprocessor and Multicomputer. Register Transfer and Micro-operation: Register Transfer Language, Register Transfer, Bus and Memory Transfers, Three State Bus Buffers, memory Transfer, Arithmetic and Logical micro-operations, Shift and Arithmetic shifts. Basic Computer Organization and Design: Instruction Codes, Stored Program Organization, Indirect Address, Computer Registers, Common Bus System, Computer Instruction, Timing and Control, Instruction Cycle, fetch Decode, Register Reference Instructions, Memory Reference Instruction, Input-Output and Interrupt, Design of Basic Computer, Design of Accumulator Logic. CPU Organization: Arithmetic and Logic Unit (ALU)- Combinational ALU, 2'S Complement Addition, Subtraction Unit, Booths Algorithm for Multiplication, Division Hardware using Restoration Division Algorithm. General register organization, Accumulator Based, Register Based, Stack Type CPU organization. Control Unit: Hardwired Control Unit, Micro-programmed Control Unit: Control memory, Address Sequencing, conditional branching, mapping of instructions, subroutine, Design of Control Unit. CPU Registers: Program Counter, Stack Pointer Register, Memory Address Register, Instruction Register, Memory Buffer Register, Flag registers, Temporary Registers. Instructions: Operational Code, Operands, Zero, One, Two and Three Address Instruction, Instruction Types, Addressing modes, Data Transfer and Manipulation instructions, Program control instructions. CISC and RISC processors: Introduction, relative merits and De-merits. Input / Output Organization: Polling, Interrupts, subroutines, Memory mapped IO, IO mapped IO, DMA, Bus Arbitration. Memory: Primary memory: ROM, PROM, EPROM, EEPROM, Flash memory, RAM: SRAM,DRAM, Cache Memory: Mapping Functions, Replacement Algorithms, Hit and Miss ratio, Virtual memories, Address Translation, Memory Management requirements, Secondary Storage: Magnetic Hard Disks.

Class	Topic
Lecture 1	Basic Structure of Computers: Basic Functional Units, Basic Operational Concept.
Lecture 2	Bus Structure, Software, Performance, Multiprocessor and Multicomputer.
Lecture 3	Register Transfer and Micro-operation: Register Transfer Language, Register Transfer, Bus and Memory Transfers.
Lecture 4	Three State Bus Buffers, memory Transfer, Arithmetic and Logical micro-operations, Shift and Arithmetic shifts.
Lecture 5	Revision on Lecturer 1 to Lecturer 4.
Lecture 6	**Class Test 1** on Lecturer 1 to Lecturer 4.
Lecture 7	Review on **Class Test 1.**
Lecture 8	Revision on **Class Test 1.**

Lecture 9	Revise Lecturer for weaker students on **Class Test 1.**
Lecture 10	Basic Computer Organization and Design: Instruction Codes, Stored Program Organization, Indirect Address.
Lecture 11	Computer Registers, Common Bus System, Computer Instruction, Timing and Control.
Lecture 12	Instruction Cycle, fetch Decode, Register Reference Instructions, Memory Reference Instruction.
Lecture 13	Input-Output and Interrupt, Design of Basic Computer, Design of Accumulator Logic. CPU Organization: Arithmetic and Logic Unit (ALU).
Lecture 14	Revision on Lecturer 10 to Lecturer 13.
Lecture 15	**Class Test 2** on Lecturer 10 to Lecturer 13.
Lecture 16	Review on **Class Test 2.**
Lecture 17	Revise Lecturer for weaker students on **Class Test 2.**
Lecture 18	Combinational ALU, 2'S Complement Addition, Subtraction Unit.
Lecture 19	Booths Algorithm for Multiplication, Division Hardware using Restoration Division Algorithm.
Lecture 20	General register organization, Accumulator Based, Register Based.
Lecture 21	Stack Type CPU organization. Control Unit: Hardwired Control Unit.
Lecture 23	Micro-programmed Control Unit: Control memory, Address Sequencing, conditional branching, mapping of instructions, subroutine.
Lecture 24	CPU Registers: Program Counter, Stack Pointer Register, Memory Address Register
Lecture 25	CPU Registers: Instruction Register, Memory Buffer Register, Flag registers, Temporary Registers.
Lecture 26	Revision on Lecturer 10 to Lecturer 13.
Lecture 27	Revision on Lecturer 10 to Lecturer 13.
Lecture 27	Revision on Lecturer 18 to Lecturer 25.
Lecture 28	Revision on Lecturer 18 to Lecturer 25.
Lecture 29	**Class Test 3** on Lecturer 10 to Lecturer 13 and Lecturer 18 to Lecturer 25.
Lecture 30	Review on **Class Test 3.**
Lecture 31	Revise Lecturer for weaker students on **Class Test 3.**
Lecture 32	Revise Lecturer for weaker students on **Class Test 3.**
Lecture 33	Instructions: Operational Code, Operands, Zero, One, Two and Three Address Instruction.
Lecture 34	Instruction Types, Brief discussion on Addressing modes.
Lecture 35	Brief discussion on Addressing modes.
Lecture 36	Data Transfer and Manipulation instructions, Program control instructions.
Lecture 37	CISC and RISC processors: Introduction, relative merits, De-merits and differences between them.
Lecture 38	Revision on Lecturer 18 to Lecturer 25.
Lecture 39	Revision on Lecturer 18 to Lecturer 25.
Lecture 40	Revision on Lecturer 33 to Lecturer 37.
Lecture 41	Revision on Lecturer 33 to Lecturer 37.
Lecture 42	**Class Test 4** on Lecturer 18 to Lecturer 25 and Lecturer 33 to Lecturer 37.

Lecture 43	Review on **Class Test 4.**
Lecture 44	Revise Lecturer for weaker students on **Class Test 4.**
Lecture 45	Revise Lecturer for weaker students on **Class Test 4.**
Lecture 46	Input / Output Organization: Polling, Interrupts, subroutines, Memory Mapped I/O, I/O Mapped I/O.
Lecture 47	DMA, Bus Arbitration. Memory: Primary memory: ROM, PROM, EPROM, EEPROM, Flash memory.
Lecture 48	RAM: SRAM, DRAM, Cache Memory: Mapping Functions, Replacement Algorithms. Differentiate between SRAM, DRAM and Cache Memory.
Lecture 49	Hit and Miss ratio, Virtual memories, Address Translation.
Lecture 50	Memory Management requirements, Secondary Storage: Magnetic Hard Disks.
Lecture 51	Revision on Lecturer 33 to Lecturer 37.
Lecture 52	Revision on Lecturer 33 to Lecturer 37.
Lecture 53	Revision on Lecturer 46 to Lecturer 50.
Lecture 54	Revision on Lecturer 46 to Lecturer 50.
Lecture 55	**Class Test 5** on Lecturer 33 to Lecturer 37 and Lecturer 46 to Lecturer 50.
Lecture 56	Review on **Class Test 5.**
Lecture 57	Revise Lecturer for weaker students on **Class Test 5.**
Lecture 58	Revise Lecturer for weaker students on **Class Test 5.**
Lecture 59	Revise Lecturer for weaker students
Lecture 60	Revise Lecturer for weaker students
Lecture 61	Revise Lecturer for weaker students
Lecture 62	Revise Lecturer for weaker students
Lecture 63	**Class Test 6** on Whole syllabus
Lecture 64	Review on **Class Test 6.**
Lecture 65	Review on **Class Test 6.**
Lecture 66	Review of **Class Test 6** and special Lecturer for weaker students.
Lecture 67	Review of **Class Test 6** and special Lecturer for weaker students.
Lecture 68	Review of **Class Test 6** and special Lecturer for weaker students.
Lecture 69	Review of **Class Test 6** and special Lecturer for weaker students.
After completion of all lecturers Students will present a Seminar on this Course	
Course Outcome:	**After completion of this course student will be able to:** Understand the working of different Sequential logic circuits. Understand working operations of different types of Flip flops as a basic building block. Know the operations of shift registers and Binary Counters. Understand the basic Computer System and general organization of different blocks. To understand the organization of memory in the Computer system and know different types of Memories.

Text/Reference Books:

1. Computer System Architecture, Morries Mano, Pearson.

2. Computer Organization & Architecture, Williams Stallings, Pearson.
3. Computer Organization, Hamacher, Vranesic and Zaky, McGraw Hill.
4. Computer Architecture and Organization, Govindrajalu, Tata McGraw Hill.
5. Computer Architecture and Organization, J P Hayes, Tata McGRaw Hill.
6. Structured Computer Organization, Andrew S. Tanenbaum, Austin, Pearson.

Head
Dept. of Computer science & BCA
Gour Mahavidyalaya
Malda

GOUR MAHAVIDYALAYA
MANGALBARI, MALDA

LESSON PLAN

PROGRAM NAME: Computer Science (Honours)

SEM: 3rd :: **PAPER:** DC6 (a)

COURSE: OPERATING SYSTEM

TEACHER NAME: Ekram Alam & Akhil Kr. Das

DC6: a) Operating System: Introduction Basic OS functions, types of operating systems: batch systems–multiprogramming systems, time sharing systems; Operating System Organization: Processor and user modes, kernels, system calls and system programs. Process System view of the process and resources, process hierarchy, threads, threading issues. Process Scheduling: Scheduling criteria, Preemptive and non-preemptive scheduling, Long term, short term and medium term, FCFS, SJF, SRTF, Priority scheduling, Round Robin, Multilevel Queue Scheduling, Multilevel Queue Feedback Scheduling. Process Synchronization: Concurrent Processes, critical section, semaphores and application, methods for inter-process communication; Deadlock: Definition, Prevention, Avoidance, Detection, Recovery, Banker's algorithm. Memory Management: Physical and virtual address space; memory allocation strategies –fixed and variable partitions, paging, segmentation, virtual memory File and I/O Management: Directory structure, file operations, file allocation methods, disc management.

CLASS	TOPIC
Lecturer 1	Introduction Basic OS functions, types of operating systems: batch systems–multiprogramming systems.
Lecturer 2	Revision on Lecture 1 and time sharing systems.
Lecturer 3	Operating System Organization: Processor and user modes, kernels.
Lecturer 4	Operating System Organization: system calls and system programs. Process System view of the process and resources.
Lecturer 5	**Class Test 1** on Lecturer 1 to Lecturer 4.
Lecturer 6	Review on **Class Test 1.**
Lecturer 7	Revision on **Class Test 1.**
Lecturer 8	Operating System Organization: Process hierarchy, threads, threading issues.
Lecturer 9	Process Scheduling: Scheduling criteria, Preemptive and non-preemptive scheduling.
Lecturer 10	Process Scheduling: Long term, short term and medium term.
Lecturer 11	**Class Test 2** on Lecturer 8 to Lecturer 10.
Lecturer 12	Review on **Class Test 2.**
Lecturer 13	Revision on **Class Test 2.**
Lecturer 14	Process Scheduling: FCFS, SJF, SRTF, Priority scheduling.
Lecturer 15	Process Scheduling: Round Robin, Multilevel Queue Scheduling, Multilevel Queue Feedback Scheduling.
Lecturer 16	Process Synchronization: Concurrent Processes, critical section.
Lecturer 17	**Class Test 3** on Lecturer 14 to Lecturer 16.
Lecturer 18	Revision on **Class Test 3.**
Lecturer 19	Revision on **Class Test 3.**
Lecturer 20	Process Synchronization: semaphores and application, methods for inter-process communication.
Lecturer 21	Deadlock: Definition, Prevention, Avoidance.
Lecturer 22	Deadlock: Detection, Recovery, Banker's algorithm.
Lecturer 23	Memory Management: Physical and virtual address space.
Lecturer 24	**Class Test 4** on Lecturer 20 to Lecturer 23.
Lecturer 25	Review on **Class Test 4.**
Lecturer 26	Revision on **Class Test 4.**
Lecturer 27	Memory Management: memory allocation strategies –fixed and variable partitions.
Lecturer 28	Memory Management: Paging, segmentation, virtual memory

Lecturer 29	File and I/O Management: Directory structure, file operations.
Lecturer 30	**Class Test 5** on Lecturer 27 to Lecturer 29.
Lecturer 31	Review on **Class Test 5.**
Lecturer 32	Revision on **Class Test 5.**
Lecturer 33	File and I/O Management: file allocation methods, disc management.
Lecturer 34	file allocation methods, disc management, Directory structure
Lecturer 35	file allocation methods, disc management, Directory structure
Lecturer 36	**Class Test 6** on Lecturer 33 to Lecturer 35.
Lecturer 37	Review on **Class Test 6.**
Lecturer 38	Revision on **Class Test 6.**
Lecturer 39	file operations & file allocation methods
Lecturer 40	file operations & file allocation methods
Lecturer 41	file operations & file allocation methods
Lecturer 42	file operations & file allocation methods
Lecturer 43	file operations & file allocation methods
Lecturer 44	**Class Test 4** on Lecturer 39 to Lecturer 43.
Lecturer 45	Review on **Class Test 4.**
Lecturer 46	Revision on **Class Test 4.**
Lecturer 47	Directory structure, file operations, file allocation methods, disc management.
Lecturer 48	Directory structure, file operations, file allocation methods, disc management.
Lecturer 49	Directory structure, file operations, file allocation methods, disc management.
Lecturer 50	**Class Test 5** on Lecturer 47 to Lecturer 49.
Lecturer 51	Review on **Class Test 5.**
Lecturer 52	Revision on **Class Test 5.**
Lecturer 53	Revise Lecturer for weaker students
Lecturer 54	Revise Lecturer for weaker students
Lecturer 55	Revise Lecturer for weaker students
Lecturer 56	**Class Test 7** on Whole syllabus
Lecturer 57	Review on **Class Test 6.**
Lecturer 58	Review on Whole syllabus
Lecturer 59	Review on Whole syllabus
Lecturer 60	Review on Whole syllabus
After completion of all lecturers Students will present a Seminar on this Course	
Course Outcome:	**After completion of this course student will be able to:** Master functions, structures and history of operating systems. Master understanding of design issues associated with operating systems. Master various process management concepts including scheduling, synchronization, and deadlocks. Be familiar with multithreading. Understand concepts of memory management including virtual memory. Master system resources sharing among the users. Master issues related to file system interface and implementation, disk management. Be familiar with protection and security mechanisms. Understand the basics of Unix system administrator.

Text/Reference Books:

1. Operating Systems Concepts, A Silberschatz, P.B. Galvin, G. Gagne, John Wiley Publications
2. Modern Operating Systems, A.S. Tanenbaum, Pearson Education
3. Operating Systems: A Modern Perspective, G. Nutt, Pearson Education.
4. Operating Systems, Internals & Design Principles W.Stallings, PHI.
5. Operating Systems- Concepts and design, M. Milenkovic, Tata McGraw Hill.
6. Sumitabha Das, UNIX Concepts and Applications, Tata McGraw-Hill.
7. D. P. Bovet and M. Cesati. Understanding the Linux Kernel, O'Reilly

GOUR MAHAVIDYALAYA MANGALBARI, MALDA

Head
Dept. of Computer science & BCA
Gour Mahavidyalaya
Malda

LESSON PLAN

PROGRAM NAME: Computer Science (Honours)

SEM: 3rd :: **PAPER:** DC6 (b)

COURSE: OPERATING SYSTEM LAB

TEACHER NAME: Ekram Alam & Akhil Kr. Das

DC6: b) Operating System Lab :	
CLASS	**TOPIC**
Lecturer 1	1. Usage of following commands: ls, pwd, tty, cat, who, who am I, rm, mkdir, rmdir, touch, cd.
Lecturer 2	2. Usage of following commands: cal, cat (append), cat (concatenate), mv, cp, man, date.
Lecturer 3	3. Usage of following commands: chmod, grep, tput (clear, highlight), bc.
Lecturer 4	4. Write a shell script to check if the number entered at the command line is prime or not.
Lecturer 5	Class Test 1 on Lecturer 1 to Lecturer 4.
Lecturer 6	Review on Class Test 1.
Lecturer 7	Revision on Class Test 1.
Lecturer 8	5. Write a shell script to modify "cal" command to display calendars of the specified months.
Lecturer 9	6. Write a shell script to modify "cal" command to display calendars of the specified range of months.
Lecturer 10	7. Write a shell script to accept a login name. If not a valid login name display message – "Entered login name is invalid".
Lecturer 11	Class Test 2 on Lecturer 9 to Lecturer 12.
Lecturer 12	Review on Class Test 2.
Lecturer 13	Revision on Class Test 2.
Lecturer 14	8. Write a shell script to display date in the mm/dd/yy format.
Lecturer 15	9. Write a shell script to display on the screen sorted output of "who" command along with the total number of users.
Lecturer 16	10. Write a shell script to display the multiplication table any number,
Lecturer 17	Class Test 3 on Lecturer 16 to Lecturer 19.
Lecturer 18	Revision on Class Test 3.
Lecturer 19	Revision on Class Test 3.
Lecturer 20	11. Write a shell script to compare two files and if found equal asks the user to delete the duplicate file.
Lecturer 21	12. Write a shell script to find the sum of digits of a given number.
Lecturer 22	13. Write a shell script to merge the contents of three files, sort the contents and then display them page by page.
Lecturer 23	14. Write a shell script to find the LCD (least common divisor) of two numbers.
Lecturer 24	Class Test 4 on Lecturer 16 to Lecturer 19.
Lecturer 25	Review on Class Test 4.
Lecturer 26	Revision on Class Test 4.
Lecturer 27	15. Write a shell script to perform the tasks of basic calculator.
Lecturer 28	16. Write a shell script to find the power of a given number.
Lecturer 29	17. Write a shell script to find the factorial of a given number.
Lecturer 30	Class Test 5 on Lecturer 16 to Lecturer 19.
Lecturer 31	Review on Class Test 5.
Lecturer 32	Revision on Class Test 5.
Lecturer 33	18. Write a shell script to check whether the number is Armstrong or not.
Lecturer 34	19. Write a shell script to check whether the file have all the permissions or not.
Lecturer 35	20. Program to show the pyramid of special character "*".
Lecturer 36	Class Test 6 on Lecturer 16 to Lecturer 19.
Lecturer 37	Review on Class Test 6.
Lecturer 38	Revision on Class Test 6.
Lecturer 39	Revise Lecturer for all students on whole syllabus for their own choice.
Lecturer 40	Revise Lecturer for all students on whole syllabus for their own choice.

Text/Reference Books:

1. Operating Systems Concepts, A Silberschatz, P.B. Galvin, G. Gagne, John Wiley Publications
2. Modern Operating Systems, A.S. Tanenbaum, Pearson Education
3. Operating Systems: A Modern Perspective, G. Nutt, Pearson Education .
4. Operating Systems, Internals & Design Principles W.Stallings, PHI.
5. Operating Systems- Concepts and design, M. Milenkovic, Tata McGraw Hill.
6. Sumitabha Das , UNIX Concepts and Applications, Tata McGraw-Hill.
7. D. P. Bovet and M. Cesati. Understanding the Linux Kernel.,O'Reilly

Head
Dept. of Computer science & BCA
Gour Mahavidyalaya
Malda

GOUR MAHAVIDYALAYA
MANGALBARI, MALDA

LESSON PLAN

PROGRAM NAME: Computer Science (Honours)

SEM: 3rd :: **PAPER:** DC7 (a)

COURSE: OBJECT ORIENTED PROGRAMMING WITH C++

TEACHER NAME: Arijit Bhattacharya

DC7: a) Object Oriented Programming with C++: Introduction to C++: Overview of Procedural Programming and Object-Orientation Programming, Using main() function, Compiling and Executing Simple Programs in C++. Concepts of Data Types, Variables, Constants, -Operators and Basic I/O Expressions, Conditional Statements and Iterative Statements, Functions and Arrays Pointers and References in C++, Memory Allocation in C++. Using Classes in C++: Principles of Object-Oriented Programming, Defining & Using Classes, Class Constructors, Constructor Overloading, Function overloading in classes, Class Variables &Functions, Objects as parameters, Specifying the Protected and Private Access, Copy Constructors, Overview of Template classes and their use. Overview of Function Overloading and Operator Overloading: Need of Overloading functions and operators, Overloading functions by number and type of arguments, Looking at an operator as a function call, Overloading Operators (including assignment operators, unary operators) Inheritance, Polymorphism and Exception Handling: Introduction to Inheritance (Multi-Level Inheritance, Multiple Inheritance), Polymorphism (Virtual Functions, Pure Virtual Functions), Abstract class, Basics Exceptional Handling (using catch and throw, multiple catch statements), Catching all exceptions, Restricting exceptions.

CLASS	TOPIC
Lecturer 1	Introduction Basic OS: functions, types of operating systems: batch systems.
Lecturer 2	Introduction Basic OS: multiprogramming systems.
Lecturer 3	Revision on Lecture 1 and time sharing systems.
Lecturer 4	**Class Test 1** on Lecturer 1 to Lecturer 4.
Lecturer 5	Review on **Class Test 1.**
Lecturer 6	Revision on **Class Test 1.**
Lecturer 7	Operating System Organization: Processor and user modes, kernels.
Lecturer 8	Operating System Organization: system calls and system programs.
Lecturer 9	Operating System Organization: Process System view of the process and resources.
Lecturer 10	**Class Test 2** on Lecturer 1 to Lecturer 4.
Lecturer 11	Review on **Class Test 2.**
Lecturer 12	Revision on **Class Test 2.**
Lecturer 13	Operating System Organization: Process hierarchy, threads.
Lecturer 14	Operating System Organization: Review Lecture 8 and threading issues.
Lecturer 15	Process Scheduling: Scheduling criteria, Preemptive and non-preemptive scheduling.
Lecturer 16	**Class Test 3** on Lecturer 9 to Lecturer 12.
Lecturer 17	Review on **Class Test 3.**
Lecturer 18	Revision on **Class Test 3.**
Lecturer 19	Process Scheduling: Long term, short term and medium term.
Lecturer 20	Process Scheduling: FCFS, SJF.
Lecturer 21	Process Scheduling: SRTF, Priority scheduling.
Lecturer 22	**Class Test 4** on Lecturer 16 to Lecturer 19.
Lecturer 23	Revision on **Class Test 4.**
Lecturer 24	Revision on **Class Test 4.**
Lecturer 25	Process Scheduling: Round Robin, Multilevel Queue Scheduling.
Lecturer 26	Process Scheduling: Multilevel Queue Feedback Scheduling.
Lecturer 27	Process Synchronization: Concurrent Processes, critical section.

Lecturer 28	**Class Test 5** on Lecturer 16 to Lecturer 19.
Lecturer 29	Revision on **Class Test 5.**
Lecturer 30	Revision on **Class Test 5.**
Lecturer 31	Process Synchronization: semaphores and application, methods for inter-process communication.
Lecturer 32	Deadlock: Definition, Prevention.
Lecturer 33	Deadlock: Avoidance, Detection.
Lecturer 34	Deadlock: Recovery, Banker's algorithm.
Lecturer 35	**Class Test 6** on Lecturer 16 to Lecturer 19.
Lecturer 36	Review on **Class Test 6.**
Lecturer 37	Revision on **Class Test 6.**
Lecturer 38	Memory Management: Physical and virtual address space.
Lecturer 39	Memory Management: memory allocation strategies –fixed and variable partitions.
Lecturer 40	Memory Management: Paging, segmentation.
Lecturer 41	**Class Test 7** on Lecturer 16 to Lecturer 19.
Lecturer 42	Review on **Class Test 7.**
Lecturer 43	Revision on **Class Test 7.**
Lecturer 44	Memory Management: Review Lecture 29 and virtual memory.
Lecturer 45	File and I/O Management: Directory structure, file operations.
Lecturer 46	File and I/O Management: file allocation methods, disc management.
Lecturer 47	**Class Test 8** on Lecturer 16 to Lecturer 19.
Lecturer 48	Review on **Class Test 8.**
Lecturer 49	Revision on **Class Test 8.**
Lecturer 50	Revision on whole syllabus.
Lecturer 51	Revision on whole syllabus.
Lecturer 52	Revision on whole syllabus.
Lecturer 53	Revision on whole syllabus.
Lecturer 54	**Class Test 9** on whole syllabus.
Lecturer 55	Review on **Class Test 9.**
Lecturer 56	Revision on **Class Test 9.**
Lecturer 57	Revision on whole syllabus for **Class Test 9.**
Lecturer 58	Revision on whole syllabus for **Class Test 9.**
Lecturer 59	Revision on whole syllabus for **Class Test 9.**
Lecturer 60	Revision on whole syllabus for **Class Test 9.**
After completion of all lecturers Students will present a Seminar on this Course	
Course Outcome:	**After completion of this course student will be able to:** Understand and can choose the loops and decision-making statements to solve the problem. Implement different Operations on arrays and will use functions to solve the given problem. Implement different Operations using pointers, ADT and file system to solve the given problem. Understand the concept of object oriented programming. Use the benefits of object oriented design and understand when it is an appropriate methodology to use. Design object oriented solutions for small systems involving multiple objects.

Text/Reference Books:

1. C++: The Complete Reference, HerbtzSchildt, McGraw Hill.
2. The C++ Programming Language, BjarneStroustrup, Addison-Wesley.
3. Programming -- Principles and Practice using C++,BjarneStroustroup, Addison-Wesley.
4. Object Oriented Programming with C++, E Balaguruswamy, Tata McGraw-Hill Education.
5. C++ How to Program, Paul Deitel, Harvey Deitel, Prentice Hall.
6. Programming with C++, John R. Hubbard, Schaum's Series.

7. Accelerated C++, Andrew Koeni, Barbara, E. Moo, Published by Addison-Wesley.

8. Effective C++, Scott Meyers, Published by Addison-Wesley.

9. Head First C++ Programming: The Definitive Beginner's Guide, Harry, H. Chaudhary, First Create space Inc, O-D Publishing, LLC USA.

10. Problem Solving with C++, Walter Savitch, Pearson Education.

11. C++ Primer, Stanley B. Lippman, Josee Lajoie, Barbara E. Moo, Published by Addison-Wesley

Head
Dept. of Computer science & BCA
Gour Mahavidyalaya
Malda

GOUR MAHAVIDYALAYA
MANGALBARI, MALDA

LESSON PLAN

PROGRAM NAME: Computer Science (Honours)

SEM: 3th :: **PAPER: DC7b**

COURSE: OBJECT ORIENTED PROGRAMMING WITH C++

TEACHER NAME: Arijit Bhattacharya

Class	Topic
Lecture 1	WAP to print the sum and product of digits of an integer
Lecture 2	WAP to reverse a number.
Lecture 3	WAP to compute the sum of the first n terms of the following series S = 1+1/2+1/3+1/4+..... WAP to compute the sum of the first n terms of the following series S =1-2+3-4+5.....
Lecture 4	Write a function that checks whether a given string is Palindrome or not. Use this function to find whether the string entered by user is Palindrome or not.
Lecture 5	Write a function to find whether a given no. is prime or not. Use the same to generate the prime numbers less than 100.
Lecture 6	WAP to compute the factors of a given number.
Lecture 7	Write a macro that swaps two numbers. WAP to use it.
Lecture 8	WAP to print a triangle of stars as follows (take number of lines from user): * *** ***** ******* *********
Lecture 9	10. WAP to perform following actions on an array entered by the user: i.Print the even-valued elements ii.Print the odd-valued elements iii.Calculate and print the sum and average of the elements of array iv.Print the maximum and minimum element of array v.Remove the duplicates from the array vi.Print the array in reverse order The program should present a menu to the user and ask for one of the options. The menu should also include options to re-enter array and to quit the program.
Lecture 10	WAP that prints a table indicating the number of occurrences of each alphabet in the text entered as command line arguments.
Lecture 11	Write a program that swaps two numbers using pointers.
Lecture 12	Write a program in which a function is passed address of two variables and then alter its contents.
Lecture 13	Write a program which takes the radius of a circle as input from the user, passes it to another function that computes the area and the circumference of the circle and displays the value of area and circumference from the main() function
Lecture 14	Revision on Lecturer 10 to Lecturer 13.
Lecture 15	Class Test 2 on Lecturer 10 to Lecturer 13.
Lecture 16	Review on Class Test 2.
Lecture 17	Revise Lecturer for weaker students on Class Test 2.
Lecture 18	Write a program to find sum of n elements entered by the user. To write this program, allocate memory dynamically using new operator.
Lecture 19	WAP to display Fibonacci series (i)using recursion, (ii) using iteration
Lecture 20	Given two ordered arrays of integers, write a program to merge the two-arrays to get an ordered array.

Lecture 21	Write a menu driven program to perform following operations on strings: a) Show address of each character in string b) Concatenate two strings without using strcat function. c) Concatenate two strings using strcat function. d) Compare two strings e) Calculate length of the string (use pointers) f) Convert all lowercase characters to uppercase g) Convert all uppercase characters to lowercase h) Calculate number of vowels i) Reverse the string
Lecture 23	WAP to calculate GCD of two numbers (i) with recursion (ii) without recursion.
Lecture 24	Create Matrix class using templates. Write a menu-driven program to perform following Matrix operations (2-D array implementation): a) Sum b) Difference c) Product d) Transpose
Lecture 25	Create the Person class. Create some objects of this class (by taking information from the user). Inherit the class Person to create two classes Teacher and Student class. Maintain the respective information in the classes and create, display and delete objects of these two classes (Use Runtime Polymorphism).
Lecture 26	Create a class Triangle. Include overloaded functions for calculating area. Overload assignment operator and equality operator.
Lecture 27	Create a class Box containing length, breath and height. Include following methods in it: a) Calculate surface Area b) Calculate Volume Write a program which takes input from the user for length, breath and height to test the above class.
Lecture 27	Create a class Box containing length, breath and height. Include following methods in it: c) Increment, Overload ++ operator (both prefix & postfix) Write a program which takes input from the user for length, breath and height to test the above class.
Lecture 28	Create a class Box containing length, breath and height. Include following methods in it: d) Decrement, Overload -- operator (both prefix & postfix) Write a program which takes input from the user for length, breath and height to test the above class.
Lecture 29	Create a class Box containing length, breath and height. Include following methods in it: e) Overload operator == (to check equality of two boxes), as a friend function Write a program which takes input from the user for length, breath and height to test the above class.
Lecture 30	Create a class Box containing length, breath and height. Include following methods in it: f) Overload Assignment operator Write a program which takes input from the user for length, breath and height to test the above class.
Lecture 31	Create a class Box containing length, breath and height. Include following methods in it: g) Check if it is a Cube or cuboid Write a program which takes input from the user for length, breath and height to test the above class.
Lecture 32	Create a structure Student containing fields for Roll No., Name, Class, Year and Total Marks. Create 10 students and store them in a file.
Lecture 33	Write a program to retrieve the student information from file created in previous question and print it in following format: Roll No. Name Marks
Lecture 34	Copy the contents of one text file to another file, after removing all whitespaces.
Lecture 35	Write a function that reverses the elements of an array in place. The function must accept only one pointer value and return void.
Lecture 36	Write a program that will read 10 integers from user and store them in an array. Implement array using pointers. The program will print the array elements in ascending and descending order.
Lecture 37	Write a program that will read 10 strings from user and store them in an array. Implement array using pointers. The program will print the array elements in ascending and descending order.
Lecture 38	Revision on Lecturer 18 to Lecturer 25.
Lecture 39	Revision on Lecturer 18 to Lecturer 25.
Lecture 40	Revision on Lecturer 33 to Lecturer 37.

GOUR MAHAVIDYALAYA
MANGALBARI, MALDA

Head
Dept. of Computer science & BCA
Gour Mahavidyalaya
Malda

LESSON PLAN

PROGRAM NAME: Computer Science (Honours)

SEM: 4th :: **PAPER:** DC 8

COURSE: THEORY OF COMPUTATION

TEACHER NAME: Deb Pratim Sinha

DC8: a) Theory of Computation : 60 hours

Finite Automata: Definition of a Finite Automaton, Model, Representation, Classification – with respect to output function Mealy and Moore Machines, with respect to State Transition – Deterministic and Non-Deterministic Machine, Examples, conversion algorithms Mealy to Moore and Moore to Mealy, Non-Deterministic to equivalent Deterministic Finite automata, Finite and Infinite state machines, Removal of Null-transitions, Acceptability of String by a Finite Automaton, Design of different Finite State Machines, Minimized Equivalent Machine.

Formal Languages and Grammar: Introduction to Formal Grammar and Language, Formal Definition, Chomsky's Classification of Grammar – Type 0, Type-1 or Context Sensitive, Type-2 or Context Free and Type-3 or Regular Grammar, Illustration of each of these classes with example, Sentential form, Sentences – Languages or strings, Derivations – left, right derivation, Derivation tree, Parse Tree, Syntax Tree, Ambiguous Grammar and Language, Designing of Grammar for a language, Finding Language for Given Grammar; Definition and basic idea about Push Down Automaton

Regular Expression: Basic Idea and Definition, Regular Expression basic Identities, Arden's Theorem and application for reduction of equivalent regular expressions, Thompson's Construction Algorithm – Regular expression to Finite Automata conversion, State Transition System to Regular Expression conversion algorithm by Arden's Algebraic Method, FA to Regular Grammar and Regular Grammar to FA conversion algorithms and applications.

Turing Machine: Concepts of Turing Machine, Formal Definitions, Classifications – Deterministic and Non-Deterministic Turing Machines, Simple Design of Turing Machines like – Unary Adder, Subtractor, Concatenator, Odd / even count etc and concepts of Universal Turing Machines.

Class	Topic
Lecture 1	Definition of a Finite Automaton, Model, Representation, Classification – with respect to output function Mealy and Moore Machines
Lecture 2	Classification – with respect to State Transition – Deterministic and Non-Deterministic Machine,
Lecture 3	Examples, conversion algorithms Mealy to Moore and Moore to Mealy
Lecture 4	Examples, conversion Non-Deterministic to equivalent Deterministic Finite automata,
Lecture 5	Revision on Lecturer 1 to Lecturer 4.
Lecture 6	Class Test 1 on Lecturer 1 to Lecturer 4.
Lecture 7	Review on **Class Test 1.**
Lecture 8	Revision on **Class Test 1.**
Lecture 9	Revise Lecturer for weaker students on **Class Test 1.**
Lecture 10	Finite and Infinite state machines, Removal of Null-transitions, Acceptability of String by a Finite Automaton
Lecture 11	Design of different Finite State Machines, Minimized Equivalent Machine
Lecture 12	Introduction to Formal Grammar and Language, Formal Definition,
Lecture 13	Chomsky's Classification of Grammar – Type 0, Type-1 or Context Sensitive, Type-2 or Context Free and Type-3 or Regular Grammar
Lecture 14	Revision on Lecturer 10 to Lecturer 13.

Lecture 15	Class Test 2 on Lecturer 10 to Lecturer 13.
Lecture 16	Review on Class Test 2.
Lecture 17	Revise Lecturer for weaker students on Class Test 2.
Lecture 18	Illustration of each of these classes with example, Sentential form, Sentences – Languages or strings
Lecture 19	Derivations – left, right derivation
Lecture 20	Derivation tree, Parse Tree, Syntax Tree
Lecture 21	Ambiguous Grammar and Language
Lecture 23	Designing of Grammar for a language
Lecture 24	Finding Language for Given Grammar
Lecture 25	Definition and basic idea about Push Down Automaton
Lecture 26	Revision on Lecturer 10 to Lecturer 13.
Lecture 27	Revision on Lecturer 10 to Lecturer 13.
Lecture 27	Revision on Lecturer 18 to Lecturer 25.
Lecture 28	Revision on Lecturer 18 to Lecturer 25.
Lecture 29	Class Test 3 on Lecturer 10 to Lecturer 13 and Lecturer 18 to Lecturer 25.
Lecture 30	Review on Class Test 3.
Lecture 31	Revise Lecturer for weaker students on Class Test 3.
Lecture 32	Revise Lecturer for weaker students on Class Test 3.
Lecture 33	Basic Idea and Definition
Lecture 34	Regular Expression basic Identities
Lecture 35	Arden's Theorem and application for reduction of equivalent regular expressions
Lecture 36	Thompson's Construction Algorithm
Lecture 37	Regular expression to Finite Automata conversion
Lecture 38	Revision on Lecturer 18 to Lecturer 25.
Lecture 39	Revision on Lecturer 18 to Lecturer 25.
Lecture 40	Revision on Lecturer 33 to Lecturer 37.
Lecture 41	Revision on Lecturer 33 to Lecturer 37.
Lecture 42	Class Test 4 on Lecturer 18 to Lecturer 25 and Lecturer 33 to Lecturer 37.
Lecture 43	Review on Class Test 4.
Lecture 44	Revise Lecturer for weaker students on Class Test 4.
Lecture 45	Revise Lecturer for weaker students on Class Test 4.
Lecture 46	State Transition System to Regular Expression conversion
Lecture 47	Algorithm by Arden's Algebraic Method
Lecture 48	FA to Regular Grammar
Lecture 49	Regular Grammar to FA conversion algorithms and applications
Lecture 50	Revision on Lecturer 33 to Lecturer 37.

Lecture 51	Revision on Lecturer 33 to Lecturer 37.
Lecture 52	Revision on Lecturer 46 to Lecturer 50.
Lecture 53	**Class Test 5** on Lecturer 33 to Lecturer 37 and Lecturer 46 to Lecturer 50.
Lecture 54	Review on **Class Test 5.**
Lecture 55	**Turing Machine:** Concepts of Turing Machine,
Lecture 56	Formal Definitions, Classifications – Deterministic and Non-Deterministic Turing Machines,
Lecture 57	Simple Design of Turing Machines like – Unary Adder, Subtractor, Concatenator,
Lecture 58	Odd / even count etc and concepts of Universal Turing Machines.
Lecture 59	**Class Test 6** on Whole syllabus
Lecture 60	Review on **Class Test 6.**
Lecture 61	Review of **Class Test 6** and special Lecturer for weaker students.
Lecture 62	Review of **Class Test 6** and special Lecturer for weaker students.
After completion of all lecturers Students will present a Seminar on this Course	
Course Outcome:	**After successfully completing this course, students will be able to:** Design a finite automaton to recognize a given regular language. Transform a language into regular expression or finite automaton or transition graph and define deterministic and nondeterministic finite automata. Prove properties of regular languages and classify them. Define relationship between regular languages and context-free grammars. Prove properties of regular languages and classify them. CO5: Building a context-free grammar for pushdown automata. Determine whether a given language is context-free language or not and Prove properties of context-free languages. Design Turing machine and Post machine for a given language. Students are exposed to a broad overview of the theoretical foundations of computer science.

Text/Reference Books:

1. *Introduction* to Automata Theory, Languages, and Computation by John E. Hopcroft,Rajeev Motwani, Jeffrey D. Ullman, 3rd Edition, Pearson.
2. Theory of Computer Science (Automata, Languages & Computation) by K L P Misra & N Chandrasekharan, PHI.
3. Introduction to Theory of Computation by Micheal Sipser, Cengage Learning.
4. Switching and Finite Automata Theory by Zvi Kohavi, Niraj.K.Jha, TMH.
5. Formal Language and Automata, P. Linz, Narosa.

D. Sinha

GOUR MAHAVIDYALAYA MANGALBARI, MALDA

Head
Dept. of Computer science & BCA
Gour Mahavidyalaya
Malda

LESSON PLAN

PROGRAM NAME: Computer Science (Honours)

SEM: 4th :: **PAPER: DC9 (a)**

COURSE: DATABSE MANAGEMENT SYSTEM

TEACHER NAME: Ekram Alam & Akhil Kr. Das

DC9: a) Database Management System :

Introduction: Drawbacks of file System; Advantages of DBMS; Layered Architecture of Database, Data Independence; Data Models; Schemas And Instances; Database Languages; Database Users, DBA; Data Dictionary; Functional Components of a DBMS.

Entity Relationship(ER) Modelling: Entity, Attributes and Relationship, Structural Constraints, Keys, ER Diagram of Some Example Database, Weak Entity Set, Specialization and Generalization, Constraints of Specialization and Generalization, Aggregation.

Relational Model: Basic Concepts of Relational Model; Relational Algebra.

Integrity Constraints: Domain Constraints, Referential Integrity, Assertions, Triggers.

Relational Database Design: Problems of Un-Normalized Database; Functional Dependencies (FD), Derivation Rules, Closure Of FD Set, Membership Of A Dependency, Canonical Cover; Decomposition to 1NF, 2NF, 3NF and BCNF Using FD; Lossless Join Decomposition Algorithm; Dependency preservation.

SQL: Basic Structure, Data Definition, Constraints and Schema Changes; Basic SQL Queries (Selection, Insertion, Deletion, Update); Order by Clause; Complex Queries, Aggregate Function and Group by Clause; Nested Sub Queries; Correlated Sub Queries; Views (Insert-Able and Updatable), Joined Relations; Set Comparisons (All, Some).

Record Storage and File Organization (Concepts only): Fixed Length and Variable Length Records; Spanned and Un-Spanned Organization of Records; Primary File Organizations and Access Structures Concepts; Unordered, Sequential, Hashed; Concepts of Primary and Secondary Index; Dense and Sparse Index; Index Sequential Files; Multilevel Indices.

Transaction Processing (Concepts only): ACID Properties; Transaction States, Concurrent Execution; Serializability (Conflict and View), Recoverability, Test for Serializability.

Class	Topic
Lecture 1	Drawbacks of file System; Advantages of DBMS
Lecture 2	Layered Architecture of Database, Data Independence
Lecture 3	Data Models; Schemas And Instances; Database Languages; Database Users, DBA
Lecture 4	Data Dictionary; Functional Components of a DBMS.
Lecture 5	Revision on Lecturer 1 to Lecturer 4.
Lecture 6	**Class Test 1** on Lecturer 1 to Lecturer 4.
Lecture 7	Review on **Class Test 1.**
Lecture 8	Revision on **Class Test 1.**
Lecture 9	Revise Lecturer for weaker students on **Class Test 1.**
Lecture 10	Entity, Attributes and Relationship, Structural Constraints, Keys, ER Diagram of Some Example Database, Weak Entity Set
Lecture 11	Specialization and Generalization, Constraints of Specialization and Generalization, Aggregation
Lecture 12	Basic Concepts of Relational Model; Relational Algebra
Lecture 13	Domain Constraints, Referential Integrity, Assertions, Triggers
Lecture 14	Revision on Lecturer 10 to Lecturer 13.
Lecture 15	**Class Test 2** on Lecturer 10 to Lecturer 13.

Lecture 16	Review on **Class Test 2.**
Lecture 17	Revise Lecturer for weaker students on **Class Test 2.**
Lecture 18	Problems of Un-Normalized Database; Functional Dependencies (FD), Derivation Rules
Lecture 19	Closure Of FD Set
Lecture 20	Membership Of A Dependency, Canonical Cover
Lecture 21	Decomposition to 1NF, 2NF, 3NF
Lecture 23	BCNF Using FD
Lecture 24	Lossless Join Decomposition Algorithm
Lecture 25	Dependency preservation
Lecture 26	Revision on Lecturer 10 to Lecturer 13.
Lecture 27	Revision on Lecturer 10 to Lecturer 13.
Lecture 27	Revision on Lecturer 18 to Lecturer 25.
Lecture 28	Revision on Lecturer 18 to Lecturer 25.
Lecture 29	**Class Test 3** on Lecturer 10 to Lecturer 13 and Lecturer 18 to Lecturer 25.
Lecture 30	Review on **Class Test 3.**
Lecture 31	Revise Lecturer for weaker students on **Class Test 3.**
Lecture 32	Revise Lecturer for weaker students on **Class Test 3.**
Lecture 33	Basic Structure, Data Definition, Constraints and Schema Changes; Basic SQL Queries (Selection, Insertion, Deletion, Update)
Lecture 34	Order by Clause; Complex Queries, Aggregate Function and Group by Clause
Lecture 35	Nested, Joined Relations
Lecture 36	Sub Queries; Correlated Sub Queries; Views (Insert-Able and Updatable)
Lecture 37	Set Comparisons (All, Some)
Lecture 38	Revision on Lecturer 18 to Lecturer 25.
Lecture 39	Revision on Lecturer 18 to Lecturer 25.
Lecture 40	Revision on Lecturer 33 to Lecturer 37.
Lecture 41	Revision on Lecturer 33 to Lecturer 37.
Lecture 42	**Class Test 4** on Lecturer 18 to Lecturer 25 and Lecturer 33 to Lecturer 37.
Lecture 43	Review on **Class Test 4.**
Lecture 44	Revise Lecturer for weaker students on **Class Test 4.**
Lecture 45	Revise Lecturer for weaker students on **Class Test 4.**
Lecture 46	Fixed Length and Variable Length Records; Spanned and Un-Spanned Organization of Records
Lecture 47	Primary File Organizations and Access Structures Concepts
Lecture 48	Unordered, Sequential, Hashed; Concepts of Primery and Secondary Index; Dense and Sparse Index
Lecture 49	Index Sequential Files; Multilevel Indices
Lecture 50	Revision on Lecturer 33 to Lecturer 37.
Lecture 51	Revision on Lecturer 33 to Lecturer 37.

Lecture 52	Revision on Lecturer 46 to Lecturer 50.
Lecture 53	**Class Test 5** on Lecturer 33 to Lecturer 37 and Lecturer 46 to Lecturer 50.
Lecture 54	Review on **Class Test 5.**
Lecture 55	ACID Properties; Transaction States, Concurrent Execution
Lecture 56	Serializability (Conflict and View), Recoverability
Lecture 57	Test for Serializability
Lecture 58	**Class Test 6** on Whole syllabus
Lecture 59	Review on **Class Test 6.**
Lecture 60	Review of **Class Test 6** and special Lecturer for weaker students.
After completion of all lecturers Students will present a Seminar on this Course	
Course Outcome:	**After successfully completing this course, students will be able to:** Understand the fundamental concepts of database. Understand user requirements and frame it in data model. Understand creations, manipulation and querying of data in databases. Solve real world problems using appropriate set, function, and relational models. Design E-R Model for given requirements and convert the same into database tables. Use SQL.

Text/Reference Books:

1. Fundamentals of Database Systems 6th Edition, R. Elmasri, S.B. Navathe, Pearson Education
2. Database Management Systems, R. Ramakrishanan, J. Gehrke, 3rd Edition, McGraw-Hill
3. Database System Concepts 6th Edition, A. Silberschatz, H.F. Korth, S. Sudarshan, McGraw Hill.
4. Database Systems Models, Languages, Design and application Programming, R. Elmasri, S.B. Navathe, Pearson Education
5. SQL and Relational Theory: How to Write Accurate SQL Code, Christopher J. Date, O'Reilly Media
6. Database Systems: A Practical Approach to Design, Implementation and Management, Thomas M. Connolly and Carolyn E. Begg, Pearson

GOUR MAHAVIDYALAYA MANGALBARI, MALDA

Head
Dept. of Computer science & BCA
Gour Mahavidyalaya
Malda

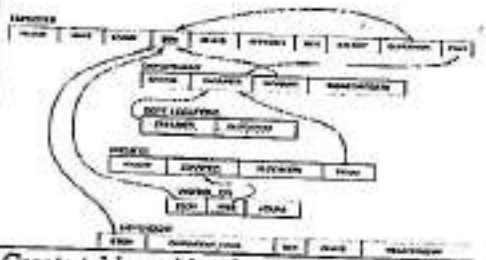
LESSON PLAN

PROGRAM NAME: Computer Science (Honours)

SEM: 4th :: **PAPER:** DC 9 (b)

COURSE: DATABSE MANAGEMENT SYSTEM

TEACHER NAME: Ekram Alam & Akhil Kr. Das

Class	Topic
Lecture 1	
	Create tables with relevant foreign key constraints and other constraints. Populate the tables with data and perform the following queries on the database
Lecture 2	Display all the details of all employees working in the company
Lecture 3	Display ssn, lname, fname, address of employees who work in department no 7.
Lecture 4	Retrieve the birthdate and address of the employee whose name is 'Franklin T. Wong'
Lecture 5	Retrieve the name and salary of every employee
Lecture 6	Retrieve all distinct salary values
Lecture 7	Retrieve all employee names whose address is in 'Bellaire
Lecture 8	Retrieve all employees who were born during the 1950s
Lecture 9	Retrieve all employees in department 5 whose salary is between 50,000 and 60,000(inclusive)
Lecture 10	Retrieve the names of all employees who do not have supervisors
Lecture 11	Retrieve SSN and department name for all employees
Lecture 12	Retrieve the name and address of all employees who work for the 'Research' department
Lecture 13	For every project located in 'Stafford', list the project number, the controlling department number, and the department manager's last name, address, and birth date.
Lecture 14	Revision on Lecturer 10 to Lecturer 13.
Lecture 15	Class Test 2 on Lecturer 10 to Lecturer 13.
Lecture 16	Review on Class Test 2.
Lecture 17	Revise Lecturer for weaker students on Class Test 2.
Lecture 18	For each employee, retrieve the employee's name, and the name of his or her immediate supervisor
Lecture 19	Retrieve all combinations of Employee Name and Department Name
Lecture 20	Make a list of all project numbers for projects that involve an employee whose last name is 'Narayan' either as a worker or as a manager of the department that controls the project.
Lecture 21	Increase the salary of all employees working on the 'ProductX' project by 15%. Retrieve employee name and increased salary of these employees.
Lecture 23	Retrieve a list of employees and the project name each works in, ordered by the employee's department, and within each department ordered alphabetically by employee first name.
Lecture 24	Select the names of employees whose salary does not match with salary of any employee in department 10

Lecture 25	Retrieve the name of each employee who has a dependent with the same first name and same sex as the employee.
Lecture 26	Retrieve the employee numbers of all employees who work on project located in Bellaire, Houston, or Stafford
Lecture 27	Find the sum of the salaries of all employees, the maximum salary, the minimum salary, and the average salary. Display with proper headings
Lecture 27	Find the sum of the salaries and number of employees of all employees of the 'Marketing' department, as well as the maximum salary, the minimum salary, and the average salary in this department.
Lecture 28	Select the names of employees whose salary is greater than the average salary of all employees in department 10
Lecture 29	For each department, retrieve the department number, the number of employees in the department, and their average salary.
Lecture 30	For each project, retrieve the project number, the project name, and the number of employees who work on that project.
Lecture 31	Change the location and controlling department number for all projects having more than 5 employees to 'Bellaire' and 6 respectively
Lecture 32	For each department having more than 10 employees, retrieve the department no, no of employees drawing more than 40,000 as salary
Lecture 33	Insert a record in Project table which violates referential integrity constraint with respect to Department number. Now remove the violation by making necessary insertion in the Department table.
Lecture 34	Delete all dependents of employee whose ssn is '123456789'
Lecture 35	Delete an employee from Employee table with ssn = '12345'(make sure that this employee has some dependents, is working on some project, is a manager of some department and is supervising some employees).
Lecture 36	Check and display the cascading effect on Dependent and Works on table. In Department table MGRSSN should be set to default value and in Employee table SUPERSSN should be set to NULL
Lecture 37	Perform a query using alter command to drop/add field and a constraint in Employee table
Lecture 38	Revision on Lecturer 18 to Lecturer 25.
Lecture 39	Revision on Lecturer 18 to Lecturer 25.
Lecture 40	Revision on Lecturer 33 to Lecturer 37.

GOUR MAHAVIDYALAYA
MANGALBARI, MALDA

Head
Dept. of Computer science & BCA
Gour Mahavidyalaya
Malda

LESSON PLAN

PROGRAM NAME: Computer Science (Honours)

SEM: 4th :: **PAPER: DC 10 (a)**

COURSE: INTRODUCTION TO MICROPROCESSOR

TEACHER NAME: Arijit Bhattacharya

DC10: a) Introduction to Microcomputer based system: Concepts of Microprocessor and Microcontrollers and their advantages and disadvantages.
Microprocessor Architecture and Memory Interfacing: Basic Architecture of Microprocessor 8085 and explanation of each block, Microprocessor 8085 pin out and signals, Addressing modes, Instruction Formats, Instruction Cycle, Clock Cycle, Multiplexed Address Data Bus, Control and Status signals, Microprocessor and Bus Timing, De-multiplexing of Address Data Bus, Generation of Control Signals for I/O and Memory, Basic concepts in Memory Interfacing, Address Decoding and memory Addresses.
Programming 8085: Instruction Set of 8085, Different Programming Techniques, Stack and Subroutines, Counter and Time Delays, Code Conversion, BCD Arithmetic and 16 bit Data Operation.
Interfacing Peripheral (I/O) and Applications: Interrupts: 8085 Interrupt, RST instructions, Software and Hardware interrupt, multiple Interrupts and Priorities, 8085 Vectored Interrupts, Restart as Software Instructions
Microprocessor 8086: Basics of 8086 microprocessor- Architecture, Instruction set, Addressing modes.

Class	Topic
Lecture 1	Concepts of Microprocessor and Microcontroller
Lecture 2	Advantages and disadvantages of Microprocessor and Microcontrollers
Lecture 3	Examples of some microprocessors and their evolution
Lecture 4	Some terminologies of microprocessor
Lecture 5	Revision on Lecturer 1 to Lecturer 4.
Lecture 6	**Class Test 1** on Lecturer 1 to Lecturer 4.
Lecture 7	Review on **Class Test 1.**
Lecture 8	Revision on **Class Test 1.**
Lecture 9	Revise Lecturer for weaker students on **Class Test 1.**
Lecture 10	Basic Architecture of Microprocessor 8085 and explanation of each block
Lecture 11	Microprocessor 8085 pin out and signals
Lecture 12	Addressing modes
Lecture 13	Instruction Cycle
Lecture 14	Revision on Lecturer 10 to Lecturer 13.
Lecture 15	**Class Test 2** on Lecturer 10 to Lecturer 13.
Lecture 16	Review on **Class Test 2.**
Lecture 17	Revise Lecturer for weaker students on **Class Test 2.**
Lecture 18	Instruction Formats
Lecture 19	Clock Cycle
Lecture 20	Multiplexed Address Data Bus
Lecture 21	Control and Status signals

Lecture 23	Microprocessor and Bus Timing
Lecture 24	De-multiplexing of Address Data Bus
Lecture 25	Generation of Control Signals
Lecture 26	Revision on Lecturer 10 to Lecturer 13.
Lecture 27	Revision on Lecturer 10 to Lecturer 13.
Lecture 27	Revision on Lecturer 18 to Lecturer 25.
Lecture 28	Revision on Lecturer 18 to Lecturer 25.
Lecture 29	**Class Test 3** on Lecturer 10 to Lecturer 13 and Lecturer 18 to Lecturer 25.
Lecture 30	Review on **Class Test 3.**
Lecture 31	Revise Lecturer for weaker students on **Class Test 3.**
Lecture 32	Revise Lecturer for weaker students on **Class Test 3.**
Lecture 33	Generation of Control Signals for I/O
Lecture 34	Generation of Control Signals for Memory
Lecture 35	Generation of Control Signals for I/O and Memory
Lecture 36	Basic concepts in Memory Interfacing
Lecture 37	Address Decoding and memory Addresses
Lecture 38	Revision on Lecturer 18 to Lecturer 25.
Lecture 39	Revision on Lecturer 18 to Lecturer 25.
Lecture 40	Revision on Lecturer 33 to Lecturer 37.
Lecture 41	Revision on Lecturer 33 to Lecturer 37.
Lecture 42	**Class Test 4** on Lecturer 18 to Lecturer 25 and Lecturer 33 to Lecturer 37.
Lecture 43	Review on **Class Test 4.**
Lecture 44	Revise Lecturer for weaker students on **Class Test 4.**
Lecture 45	Revise Lecturer for weaker students on **Class Test 4.**
Lecture 46	Instruction Set of 8085, Different Programming Techniques
Lecture 47	Stack and Subroutines, Counter and Time Delays
Lecture 48	Code Conversion, BCD Arithmetic and 16 bit Data Operation
Lecture 49	Interrupts: 8085 Interrupt, RST instructions, Software and Hardware interrupt,
Lecture 50	multiple Interrupts and Priorities, 8085 Vectored Interrupts
Lecture 51	Restart as Software Instructions
Lecture 52	Revision on Lecturer 46 to Lecturer 50.
Lecture 53	**Class Test 5** on Lecturer 33 to Lecturer 37 and Lecturer 46 to Lecturer 50.
Lecture 54	Review on **Class Test 5.**
Lecture 55	Basics of 8086 microprocessor- Architecture
Lecture 56	Instruction set
Lecture 57	Addressing modes

Lecture 58	Class Test 6 on Whole syllabus
Lecture 59	Review on Class Test 6.
Lecture 60	Review of Class Test 6 and special Lecturer for weaker students.
After completion of all lecturers Students will present a Seminar on this Course	
Course Outcome:	**After the completion of the course the student will be able to:** Analyze and design various microprocessor types and their characteristics. CO2: To evaluate several applications of Microprocessor. Develop practical understanding, limitations and constraints of the theory they learn. Understand the architecture of 8085 and 8051. Impart the knowledge about the instruction set.

Text/Reference Books:

1. Microprocessor architecture, programming and applications with 8085/8085A, Wiley eastern Ltd, by Ramesh S. Gaonkar.
2. Intel Corp: The 8085 / 8085A. Microprocessor Book – Intel marketing communication, Wiley inter science publications.
3. An introduction to micro computers Vol. 2 – some real Microprocessor – Galgotia Book Source, New Delhi by Adam Osborne and J. Kane.
4. Advanced Microprocessors by Ray and Bhurchandi - TMH.
5. Intel Corp. Micro Controller Handbook – Intel Publications.
6. Microprocessors and Interfacing by Douglas V. Hall, McGraw Hill International
7. Assembly Language Programming the IBM PC by Alan R. Miller, Subex Inc.
8. The Intel Microprocessors: 8086/8088, 80186, 80286, 80386 & 80486, Bary B. Brey, Prentice Hall, India

GOUR MAHAVIDYALAYA MANGALBARI, MALDA

Head
Dept. of Computer science & BCA
Gour Mahavidyalaya
Malda

LESSON PLAN

PROGRAM NAME: Computer Science (Honours)

SEM: 5th :: **PAPER: DC 11**

COURSE: DATA COMMUNICATION & NETWORKING

TEACHER NAME: Dr. Subhendu Chatterjee

DC11: Data Communication & Networking: Data Communication Concepts: Analog and Digital Signals, Periodic and Non-periodic signals, Time and Frequency Domain, Bandwidth and Data rate, Signal rate, Serial and Parallel Transmission. Protocol. Various modes of transmission: Simplex/ Half Duplex, Duplex; Features of guided and unguided transmission media; Circuit Switching, Packet Switching; transmission impairment. Physical structure of Network: Types of connections (Topologies), Categories of Computer Network: LAN, MAN, WAN; Digital to Digital conversion: line coding schemes; Analog to Digital Conversion: PCM, DM; Digital to Analog conversion: ASK, PSK, FSK, QAM; Modulation and Encoding: AM, FM, PM; Multiplexing: FDM, TDM, WDM; OSI &TCP/IP Model. Error detection and correction: CRC, Checksum, Hamming Code; Protocols: IP, ARP, RARP, TCP, UDP, SMTP, FTP, DNS, DHCP etc.

Class	Topic
Lecture 1	What is data communication and its different aspects?
Lecture 2	What is network? Why we use networks, basic requirements.
Lecture 3	Data Communication Concepts: Analog and Digital Signals
Lecture 4	Periodic and Non-periodic signals
Lecture 5	Time and Frequency Domain, Bandwidth and Data rate, Signal rate
Lecture 6	Serial and Parallel Transmission. Protocol.
Lecture 7	Revise Lecturer of Lecturer 1 to Lecturer 6
Lecture 8	**Class Test 1** From Lecturer 1 to Lecturer 6
Lecture 9	Review of **Class Test 1**
Lecture 10	Revise Lecturer for weaker students on **Class Test 1.**
Lecture 11	Various modes of transmission: Simplex, Half Duplex, Duplex.
Lecture 12	Features of guided and unguided transmission media
Lecture 13	Circuit Switching, Packet Switching. Differentiate Between Circuit Switching and Packet Switching
Lecture 14	Transmission impairment. Physical structure of Network: Types of connections (Topologies).
Lecture 15	Revise Lecturer of Lecturer 1 to Lecturer 6
Lecture 16	Revise Lecturer of Lecturer 1 to Lecturer 6
Lecture 17	Revise Lecturer of Lecturer 11 to Lecturer 14
Lecture 18	Revise Lecturer of Lecturer 11 to Lecturer 14

Lecture 19	**Class Test 2** From Lecturer 1 to Lecturer 6 and Lecturer 11 to Lecturer 14
Lecture 20	Review of **Class Test 2**
Lecture 21	Revise Lecturer for weaker students on **Class Test 2.**
Lecture 23	Categories of Computer Network: LAN, MAN, WAN.
Lecture 24	Revise Lecturer 23 and Differentiate Between LAN, MAN and WAN.
Lecture 25	Digital to Digital conversion: line coding schemes; Analog to Digital Conversion: PCM, DM.
Lecture 26	Digital to Analog conversion: ASK, PSK, FSK, and QAM.
Lecture 27	Modulation and Encoding: AM, FM, PM
Lecture 27	Revise Lecturer of Lecturer 1 to Lecturer 6
Lecture 28	Revise Lecturer of Lecturer 1 to Lecturer 6
Lecture 29	Revise Lecturer of Lecturer 11 to Lecturer 14
Lecture 30	Revise Lecturer of Lecturer 11 to Lecturer 14
Lecture 31	Revise Lecturer of Lecturer 23 to Lecturer 27
Lecture 32	Revise Lecturer of Lecturer 23 to Lecturer 27
Lecture 33	**Class Test 3** From Lecturer 11 to Lecturer 14 and Lecturer 23 to Lecturer 27
Lecture 34	Review of **Class Test 3.**
Lecture 35	Revise Lecturer for weaker students on **Class Test 3.**
Lecture 36	Revise Lecturer for weaker students on **Class Test 3.**
Lecture 37	Multiplexing: FDM, TDM, WDM
Lecture 38	Brief Discussion of OSI model.
Lecture 39	Brief Discussion of OSI model.
Lecture 40	Brief Discussion of TCP/IP Model
Lecture 41	Brief Discussion of TCP/IP Model. Differentiate Between OSI model and TCP/IP Model.
Lecture 42	Revise Lecturer of Lecturer 23 to Lecturer 27
Lecture 43	Revise Lecturer of Lecturer 23 to Lecturer 27
Lecture 44	Revise Lecturer of Lecturer 37 to Lecturer 41
Lecture 45	Revise Lecturer of Lecturer 37 to Lecturer 41
Lecture 46	**Class Test 4** From Lecturer 23 to Lecturer 27 and Lecturer 37 to Lecturer 41
Lecture 47	Review of **Class Test 4.**
Lecture 48	Revise Lecturer weaker students on **Class Test 4.**
Lecture 49	Revise Lecturer weaker students on **Class Test 4.**
Lecture 50	Error detection and correction: CRC, Checksum, Hamming Code.
Lecture 51	Error detection and correction: CRC, Checksum, Hamming Code.
Lecture 52	Protocols: IP, ARP, RARP, TCP.
Lecture 53	Protocols: UDP, SMTP, FTP, DNS, DHCP etc.
Lecture 54	Revise Lecturer From Lecturer 50 to Lecturer 51.

Lecture 55	Revise Lecturer From Lecturer 50 to Lecturer 51.
Lecture 56	Revise Lecturer From Lecturer 52 to Lecturer 53.
Lecture 57	Revise Lecturer From Lecturer 52 to Lecturer 53.
Lecture 58	Revise Lecturer of Lecturer 37 to Lecturer 41
Lecture 59	Revise Lecturer of Lecturer 37 to Lecturer 41
Lecture 60	**Class Test 5** From Lecturer 37 to Lecturer 41 and Lecturer 50 to Lecturer 53
Lecture 61	Review of **Class Test 5.**
Lecture 62	Revise Lecturer weaker students on **Class Test 5.**
Lecture 63	Revise Lecturer for weaker students on whole syllabus.
Lecture 64	Revise Lecturer for weaker students on whole syllabus.
Lecture 65	Revise Lecturer for weaker students on whole syllabus.
Lecture 66	**Class Test 6** on whole syllabus.
Lecture 67	Review of **Class Test 6** and special lecturer for weaker students
Lecture 68	Review of **Class Test 6** and special lecturer for weaker students
Lecture 69	Review of **Class Test 6** and special lecturer for weaker students
After completion of all lecturers Students will present a Seminar on this Course	
Course Outcome:	**After successfully completing this course, students will be able to:** Students will get acquainted with fundamentals of Networking like PAN, LAN, MAN, WAN, topologies and Home & Business applications of Networks. Students will clear their basic concepts about the standards, their need & types of standards. Students will know the design issues for the layers, layered architecture of the Network Models & functions performed at each layer. CO4: Students will come to know the role played by different addresses at different layers of the network models. Students will understand very basic networking hardware like transmission media types & tools description. Students will be able to understand the need and importance of protocols at each layer in the communicating computers.

Text/Reference books:

1. Data Communications and Networking, B. A. Forouzan, THM.
2. Computer Networks, A.S. Tanenbaum, PHI.
3. Data and Computer Communication, W. Stallings, PHI/ Pearson Education
4. Data & Computer Communication, Black, PHI.

Chatterjee

GOUR MAHAVIDYALAYA MANGALBARI, MALDA

Head
Dept. of Computer science & BCA
Gour Mahavidyalaya
Malda

LESSON PLAN

PROGRAM NAME: Computer Science (Honours)

SEM: 5th :: **PAPER:** DC12 (a)

COURSE: COMPUTER GRAPHICS

TEACHER NAME: Deb Pratim Sinha

DC12: a) Computer Graphics

Introduction: Basic concepts of Graphics Devices– Monochrome and Color Monitor displaying technique only, Physical and logical units of graphics devices –Pixel and its different properties, Basic idea for image or picture formation using pixels –Raster Scan and Vector Scan, Image Color Model, Color Coding, Lookup Table based color mapping.

Basic geometrical shapes formation algorithms: Concepts Co-ordinate System, Line Segment, Circle, elliptic segment and its formation; DDA, Bresenham's and Midpoint scan conversion algorithms.

Two and Three Dimensional Transformations: Geometric Transformations operations - Translation, Rotation, Scaling. Reflection, Shearing, Homogeneous coordinate system representation, matrix representation Coordinate Transformations operations - Translation, Rotation, Scaling. Reflection, Shearing, Composite Transformations Operations – Basic ideas and matrix representations by matrix concatenation for a particular operation.

Clipping: Point Clipping, Line Clipping – Region coding, Cohen-Sutherland Algorithm;

Area filling: Boundary fill and flood fill

Projection: Basic Concept of Projection operation and its application, Classification – Perspective, Parallel.

Applications: Basic Concepts Computer Art – publishing, drawing and drafting, Animation – Animating and modelling of real world, Morphing – Classification of morphing and Application to the Advertisements and publicities.

Class	Topic
Lecture 1	Introduction of basic concepts of Graphics Display Devices
Lecture 2	Display techniques of Monochrome and Color Monitor
Lecture 3	Physical and logical units of graphics devices –Pixel and its different properties
Lecture 4	Concept of pixel and its different properties
Lecture 5	Revision on Lecturer 1 to Lecturer 4.
Lecture 6	**Class Test 1** on Lecturer 1 to Lecturer 4.
Lecture 7	Review on **Class Test 1.**
Lecture 8	Revision on **Class Test 1.**
Lecture 9	Revise Lecturer for weaker students on **Class Test 1.**
Lecture 10	Basic idea for image or picture formation using pixels
Lecture 11	Raster Scan and Vector Scan
Lecture 12	Image Color Model, Color Coding
Lecture 13	Lookup Table based color mapping
Lecture 14	Revision on Lecturer 10 to Lecturer 13.
Lecture 15	**Class Test 2** on Lecturer 10 to Lecturer 13.
Lecture 16	Review on **Class Test 2.**
Lecture 17	Revise Lecturer for weaker students on **Class Test 2.**

Lecture 18	Concepts Co-ordinate System
Lecture 19	Line Segment, Circle and its formation
Lecture 20	Line Segment, Circle, elliptic segment and its formation
Lecture 21	Concept of Two dimensional geometry for line drawing using computer
Lecture 23	DDA, Bresenham's and Midpoint scan conversion algorithms
Lecture 24	Geometric Transformations operations - Translation, Rotation, Scaling. Reflection, Shearing
Lecture 25	Homogeneous coordinate system representation, matrix representation
Lecture 26	Revision on Lecturer 10 to Lecturer 13.
Lecture 27	Revision on Lecturer 10 to Lecturer 13.
Lecture 27	Revision on Lecturer 18 to Lecturer 25.
Lecture 28	Revision on Lecturer 18 to Lecturer 25.
Lecture 29	**Class Test** 3 on Lecturer 10 to Lecturer 13 and Lecturer 18 to Lecturer 25.
Lecture 30	Review on **Class Test 3.**
Lecture 31	Revise Lecturer for weaker students on **Class Test** 3.
Lecture 32	Revise Lecturer for weaker students on **Class Test** 3.
Lecture 33	Coordinate Transformations operations - Translation, Rotation
Lecture 34	Coordinate Transformations operations - Scaling. Reflection, Shearing
Lecture 35	Composite Transformations Operations
Lecture 36	Basic ideas and matrix representations by matrix concatenation for a particular operation
Lecture 37	Point Clipping, Line Clipping – Region coding, Cohen-Sutherland Algorithm
Lecture 38	Revision on Lecturer 18 to Lecturer 25.
Lecture 39	Revision on Lecturer 18 to Lecturer 25.
Lecture 40	Revision on Lecturer 33 to Lecturer 37.
Lecture 41	Revision on Lecturer 33 to Lecturer 37.
Lecture 42	**Class Test** 4 on Lecturer 18 to Lecturer 25 and Lecturer 33 to Lecturer 37.
Lecture 43	Review on **Class Test 4.**
Lecture 44	Revise Lecturer for weaker students on **Class Test** 4.
Lecture 45	Revise Lecturer for weaker students on **Class Test** 4.
Lecture 46	**Area filling:** Boundary fill
Lecture 47	**Area filling:** flood fill
Lecture 48	**Projection:** Basic Concept of Projection operation and its application,
Lecture 49	Classification – Perspective, Parallel.
Lecture 50	Revision on Lecturer 33 to Lecturer 37.
Lecture 51	Revision on Lecturer 33 to Lecturer 37.
Lecture 52	Revision on Lecturer 46 to Lecturer 50.
Lecture 53	**Class Test** 5 on Lecturer 33 to Lecturer 37 and Lecturer 46 to Lecturer 50.

Lecture 54	Review on **Class Test 5.**
Lecture 55	Basic Concepts Computer Art – publishing, drawing and drafting
Lecture 56	Animation – Animating and modelling of real world
Lecture 57	Morphing – Classification of morphing and Application to the Advertisements and publicities
Lecture 58	**Class Test 6** on Whole syllabus
Lecture 59	Review on **Class Test 6.**
Lecture 60	Review of **Class Test 6** and special Lecturer for weaker students.
After completion of all lecturers Students will present a Seminar on this Course	
Course Outcome:	**After successfully completing this course, students will be able to:** Understand how to use graphics objects represented in computer. Will able to correlate between user and computer through graphics. Able to increase the productivity through graphics. Understand programmer's perspective of working of computer graphics. Compare various graphics algorithm used in 2D and 3D Be able to understand fundamentals of graphics used in various real life applications. Understand and identify the performance characteristics of graphics algorithms.

Text/Reference Books:

1. Computer Graphics by Zhigang Xiang, Roy Plastock, Schaum's Outlines Series
2. Computer Graphics by Hern & Baker
3. Procedural Elements for Computer Graphics by David F. Roger, 2nd Edition, TMH
4. Computer Graphics by Folly & Vandam

D. Sinha

GOUR MAHAVIDYALAYA
MANGALBARI, MALDA

Head
Dept. of Computer science & BCA
Gour Mahavidyalaya
Malda

LESSON PLAN

PROGRAM NAME: Computer Science (Honours)

SEM: 5th :: **PAPER:** DSE1 E2(a)

COURSE: INTRODUCTION TO PYTHON PROGRAMMING

TEACHER NAME: Arijit Bhattacharya & Ekram Alam

DSE1 E2(a) : Introduction to Python Programming

Overview of Programming: Structure of a Python Program, Elements of Python

Introduction to Python: Python Interpreter, Using Python as calculator, Python shell, Indentation, Atoms, Identifiers and keywords, Literals, Strings, Operators(Arithmetic operator, Relational operator, Logical or Boolean operator, Assignment, Operator, Ternary operator, Bit wise operator)

Creating Python Programs: Input and Output Statements, Control statements (Branching, Looping, Conditional Statement, Exit function, Difference between break, continue and pass.), Defining Functions, default arguments, Exception handling.

Iteration and Recursion: Conditional execution, Alternative execution, Nested conditionals, Return statement, Recursion, Stack diagrams for recursive functions, Multiple assignment, while statement, for statement.

Strings and Lists: String as a compound data type, Length, Traversal and the for loop, String slices, String comparison, A find function, Looping and counting, List values, Accessing elements, List length, List membership, Lists and for loops, List operations, List deletion; Nested lists.

Object Oriented Programming: Introduction to Classes, Objects and Methods, Standard Libraries

Class	Topic
Lecture 1	Structure of a Python Program
Lecture 2	Elements of Python Program
Lecture 3	Python Interpreter , Using Python as calculator
Lecture 4	Python shell, Indentation
Lecture 5	Revision on Lecturer 1 to Lecturer 4.
Lecture 6	**Class Test 1** on Lecturer 1 to Lecturer 4.
Lecture 7	Review on **Class Test 1.**
Lecture 8	Revision on **Class Test 1.**
Lecture 9	Revise Lecturer for weaker students on **Class Test 1.**
Lecture 10	Atoms, Identifiers and keywords, Literals, Strings
Lecture 11	Assignment Operator, Ternary operator, Bit wise operator Operators
Lecture 12	Arithmetic operator, Relational operator
Lecture 13	Logical or Boolean operator
Lecture 14	Revision on Lecturer 10 to Lecturer 13.
Lecture 15	**Class Test 2** on Lecturer 10 to Lecturer 13.
Lecture 16	Review on **Class Test 2.**
Lecture 17	Revise Lecturer for weaker students on **Class Test 2.**
Lecture 18	Input and Output Statements, Exit function
Lecture 19	Control statements : Looping, Conditional Statement,

Lecture 20	Control statements : Conditional Statement
Lecture 21	Difference between break, continue and pass
Lecture 23	Defining Functions
Lecture 24	Default arguments
Lecture 25	Exception handling
Lecture 26	Revision on Lecturer 10 to Lecturer 13.
Lecture 27	Revision on Lecturer 10 to Lecturer 13.
Lecture 27	Revision on Lecturer 18 to Lecturer 25.
Lecture 28	Revision on Lecturer 18 to Lecturer 25.
Lecture 29	**Class Test 3** on Lecturer 10 to Lecturer 13 and Lecturer 18 to Lecturer 25.
Lecture 30	Review on **Class Test 3.**
Lecture 31	Revise Lecturer for weaker students on **Class Test 3.**
Lecture 32	Revise Lecturer for weaker students on **Class Test 3.**
Lecture 33	Conditional execution, Alternative execution,
Lecture 34	Nested conditionals, Return statement
Lecture 35	Recursion, Stack diagrams for recursive functions
Lecture 36	Multiple assignment
Lecture 37	Looping : while statement, for statement
Lecture 38	Revision on Lecturer 18 to Lecturer 25.
Lecture 39	Revision on Lecturer 18 to Lecturer 25.
Lecture 40	Revision on Lecturer 33 to Lecturer 37.
Lecture 41	Revision on Lecturer 33 to Lecturer 37.
Lecture 42	**Class Test 4** on Lecturer 18 to Lecturer 25 and Lecturer 33 to Lecturer 37.
Lecture 43	Review on **Class Test 4.**
Lecture 44	Revise Lecturer for weaker students on **Class Test 4.**
Lecture 45	Revise Lecturer for weaker students on **Class Test 4.**
Lecture 46	String as a compound data type, Length, Traversal and the for loop
Lecture 47	String slices, String comparison, A find function
Lecture 48	Looping and counting, List values, Accessing elements, List length, List membership, Lists and for loops
Lecture 49	List operations, List deletion; Nested lists
Lecture 50	Revision on Lecturer 33 to Lecturer 37.
Lecture 51	Revision on Lecturer 33 to Lecturer 37.
Lecture 52	Revision on Lecturer 46 to Lecturer 50.
Lecture 53	**Class Test 5** on Lecturer 33 to Lecturer 37 and Lecturer 46 to Lecturer 50.
Lecture 54	Review on **Class Test 5.**
Lecture 55	Introduction to Classes

Lecture 56	Objects and Methods
Lecture 57	Standard Libraries
Lecture 58	**Class Test 6** on Whole syllabus
Lecture 59	Review on **Class Test 6.**
Lecture 60	Review of **Class Test 6** and special Lecturer for weaker students.
After completion of all lecturers Students will present a Seminar on this Course	
Course Outcome:	After successfully completing this course, students will be able to: Understand why Python is a useful scripting language for developers. Learn how to use lists, tuples, and dictionaries in Python programs. Learn how to identify Python object types. Learn how to use indexing and slicing to access data in Python programs. Define the structure and components of a Python program. Learn how to write loops and decision statements in Python. Learn how to write functions and pass arguments in Python. Learn how to build and package Python modules for reusability. Learn how to design object-oriented programs with Python classes.

Text/Reference Books:

1. Introduction to Computation and Programming Using Python, John V. Guttag, MIT Press
2. Think Python: How to Think Like a Computer Scientist, Allen Downey, O'Reilly
3. Learning Python, Mark Lutz, O'Reilly
4. Python Programming for the Absolute Beginner, Michael Dawson, Cengage Learning
5. Learning to Program in Python, P. M. Heathcote, PG Online Limited
6. Python Programming Fundamentals, Authors: Lee and Kent D.

Head
Dept. of Computer science & BCA
Gour Mahavidyalaya
Malda

GOUR MAHAVIDYALAYA
MANGALBARI, MALDA

LESSON PLAN

PROGRAM NAME: Computer Science (Honours)

SEM: 5th :: **PAPER: DSE 2-E2**

COURSE: INTELLIGENT SYSTEMS

TEACHER NAME: Akhil Kr. Das

DC14 : Intelligent System

Introduction: Introduction to Artificial Intelligence, Background and Applications, Turing Test and Rational Agent approaches to AI, Introduction to Intelligent Agents, their structure, behavior and environment.

Problem Solving and Searching Techniques: Problem Characteristics, Production Systems, Control Strategies, Breadth First Search, Depth First Search, Hill climbing and its Variations, Heuristics Search Techniques: Best First Search, A* algorithm, Constraint Satisfaction Problem, Means-End Analysis, Introduction to Game Playing, Min-Max and Alpha-Beta pruning algorithms.

Knowledge Representation: Introduction to First Order Predicate Logic, Resolution Principle, Unification, Semantic Nets, Conceptual Dependencies, Frames, and Scripts, Production Rules, Conceptual Graphs. Programming in Logic (PROLOG)

Dealing with Uncertainty and Inconsistencies: Truth Maintenance System, Default Reasoning, Probabilistic Reasoning, Bayesian Probabilistic Inference, Possible World Representations.

Class	Topic
Lecture 1	Introduction to Artificial Intelligence
Lecture 2	Introduction to Artificial Intelligence – continued
Lecture 3	Background of Artificial Intelligence
Lecture 4	Applications of Artificial Intelligence
Lecture 5	Revision on Lecturer 1 to Lecturer 4.
Lecture 6	**Class Test 1** on Lecturer 1 to Lecturer 4.
Lecture 7	Review on **Class Test 1.**
Lecture 8	Revision on **Class Test 1.**
Lecture 9	Revise Lecturer for weaker students on **Class Test 1.**
Lecture 10	Turing Test and Rational Agent approaches to AI,
Lecture 11	Introduction to Intelligent Agents, their structure
Lecture 12	Introduction to Intelligent Agents, their behavior
Lecture 13	Introduction to Intelligent Agents, their environment
Lecture 14	Revision on Lecturer 10 to Lecturer 13.
Lecture 15	**Class Test 2** on Lecturer 10 to Lecturer 13.
Lecture 16	Review on **Class Test 2.**
Lecture 17	Revise Lecturer for weaker students on **Class Test 2.**
Lecture 18	Problem Solving and Searching Techniques: Problem Characteristics
Lecture 19	Production Systems, Control Strategies
Lecture 20	Breadth First Search, Depth First Search,
Lecture 21	Hill climbing and its Variations, Heuristics Analysis

Lecture 23	Search Techniques: Best First Search, A* algorithm
Lecture 24	Constraint Satisfaction Problem, Means-End
Lecture 25	Introduction to Game Playing, Min-Max and Alpha-Beta pruning algorithms
Lecture 26	Revision on Lecturer 10 to Lecturer 13.
Lecture 27	Revision on Lecturer 10 to Lecturer 13.
Lecture 27	Revision on Lecturer 18 to Lecturer 25.
Lecture 28	Revision on Lecturer 18 to Lecturer 25.
Lecture 29	**Class Test 3** on Lecturer 10 to Lecturer 13 and Lecturer 18 to Lecturer 25.
Lecture 30	Review on **Class Test 3.**
Lecture 31	Revise Lecturer for weaker students on **Class Test 3.**
Lecture 32	Revise Lecturer for weaker students on **Class Test 3.**
Lecture 33	Knowledge Representation: Introduction to First Order Predicate Logic
Lecture 34	Knowledge Representation: Introduction to First Order Predicate Logic – continued
Lecture 35	Resolution Principle
Lecture 36	Unification, Semantic Nets, Conceptual Dependencies
Lecture 37	Frames, and Scripts, Production Rules
Lecture 38	Revision on Lecturer 18 to Lecturer 25.
Lecture 39	Revision on Lecturer 18 to Lecturer 25.
Lecture 40	Revision on Lecturer 33 to Lecturer 37.
Lecture 41	Revision on Lecturer 33 to Lecturer 37.
Lecture 42	**Class Test 4** on Lecturer 18 to Lecturer 25 and Lecturer 33 to Lecturer 37.
Lecture 43	Review on **Class Test 4.**
Lecture 44	Revise Lecturer for weaker students on **Class Test 4.**
Lecture 45	Revise Lecturer for weaker students on **Class Test 4.**
Lecture 46	Conceptual Graphs. Programming in Logic (PROLOG)
Lecture 47	Conceptual Graphs. Programming in Logic (PROLOG) – continued
Lecture 48	Dealing with Uncertainty and Inconsistencies:
Lecture 49	Truth Maintenance System
Lecture 50	Revision on Lecturer 33 to Lecturer 37.
Lecture 51	Revision on Lecturer 33 to Lecturer 37.
Lecture 52	Revision on Lecturer 46 to Lecturer 50.
Lecture 53	**Class Test 5** on Lecturer 33 to Lecturer 37 and Lecturer 46 to Lecturer 50.
Lecture 54	Review on **Class Test 5.**
Lecture 55	Default Reasoning , Probabilistic Reasoning
Lecture 56	Bayesian Probabilistic Inference
Lecture 57	Possible World Representations

Lecture 58	Class Test 6 on Whole syllabus
Lecture 59	Review on Class Test 6.
Lecture 60	Review of Class Test 6 and special Lecturer for weaker students.
After completion of all lecturers Students will present a Seminar on this Course	
Course Outcome:	After undergoing the course, Students will be able to: Learn the use of AI in different real life problems. Use the heuristic search techniques for AI related problems. Analyze and formalize the problem (as a state space, graph, etc.) and select the appropriate search method. Choose an appropriate problem-solving method. Know how knowledge is represented in computer system and different knowledge-representation scheme.

Text/Reference Books:

1. Introduction to A.I and Expert Systems, DAN.W Patterson– PHI
2. Artificial Intelligence-A Modern Approach Russell &Norvig, ,LPE, Pearson Prentice Hall
3. Artificial Intelligence, Rich & Knight,– Tata McGraw Hill
4. Programming in PROLOG ,W.F. Clocksin and Mellish, Narosa Publishing Hous
5. Prolog Programming for Artificial Intelligence ,IvanBratko, Addison-Wesley, Pearson Education

GOUR MAHAVIDYALAYA
MANGALBARI, MALDA

Head
Dept. of Computer science & BCA
Gour Mahavidyalaya
Malda

LESSON PLAN

PROGRAM NAME: Computer Science (Honours)

SEM: 5th :: **PAPER: SEC-1**

COURSE: SENSOR NETWORK & IOT

TEACHER NAME: Dipanjan Saha & Shib Charan Chowdhury

SEC-1: Sensor Network & IOT: IoT Architecture-State of the Art – Introduction, State of the art Architecture Reference Model- Introduction, Reference Model and architecture, IoT reference Model. M2M and IoT Technology Fundamentals- Devices and gateways, Local and wide area networking, Data management, Business processes in IoT, Everything as a Service (XaaS), M2M and IoT Analytics, Knowledge Management. M2M to IoT – A Market Perspective– Introduction, Some Definitions, M2M Value Chains, IoT Value Chains, An emerging industrial structure for IoT, The international driven global value chain and global information monopolies. M2M to IoT-An Architectural Overview– Building architecture, Main design principles and needed capabilities, An IoT architecture outline, standards considerations. IoT Reference Architecture- Introduction, Functional View, Information View, Deployment and Operational View, Other Relevant architectural views. Real-World Design Constraints- Introduction, Technical Design constraints hardware is popular again, Data representation and visualization, Interaction and remote control. Industrial Automation- Service-oriented architecture based device integration, SOCRADES: realizing the enterprise integrated Web of Things, IMC-AESOP: from the Web of Things to the Cloud of Things.

CLASSS	TOPIC
Lecturer 1	IoT Architecture-State of the Art – Introduction.
Lecturer 2	State of the art Architecture Reference Model: Introduction.
Lecturer 3	State of the art Architecture Reference Model: Reference Model and architecture.
Lecturer 4	State of the art Architecture Reference Model: IoT reference Model.
Lecturer 5	Revision Class from Lecturer 1 to Lecturer 4
Lecturer 6	Class Test 1 on Lecturer 1 to lecturer 4.
Lecturer 7	Review on Class Test 1.
Lecturer 8	Special Class for Weaker Students on Class Test 1.
Lecturer 9	Special Class for Weaker Students on Class Test 1.
Lecturer 10	Special Class for all Students on Class Test 1.
Lecturer 11	M2M and IoT Technology Fundamentals: Devices and gateways, Local and wide area networking.
Lecturer 12	M2M and IoT Technology Fundamentals: Data management, Business processes in IoT, Everything as a Service (XaaS).
Lecturer 13	M2M and IoT Technology Fundamentals: M2M and IoT Analytics, Knowledge Management.
Lecturer 14	Revision Class from Lecturer 1 to Lecturer 4.
Lecturer 15	Revision Class from Lecturer 1 to Lecturer 4.
Lecturer 16	Revision Class from Lecturer 11 to Lecturer 13.
Lecturer 17	Class Test 2 on Lecturer 1 to lecturer 4 and Lecturer 11 to Lecturer 13.
Lecturer 18	Review on Class Test 2.
Lecturer 19	Special Class for Weaker Students on Class Test 2.
Lecturer 20	Special Class for Weaker Students on Class Test 2.

Lecturer 21	Special Class for all Students on **Class Test 2.**
Lecturer 22	M2M to IoT – A Market Perspective: Introduction, Some Definitions, M2M Value Chains.
Lecturer 23	M2M to IoT – A Market Perspective: IoT Value Chains, An emerging industrial structure for IoT.
Lecturer 24	M2M to IoT – A Market Perspective: The international driven global value chain and global information monopolies.
Lecturer 25	Revision Class from Lecturer 22 to Lecturer 24
Lecturer 26	**Class Test** 3 on Lecturer 22 to lecturer 24.
Lecturer 27	Review on **Class Test 3.**
Lecturer 28	Special Class for Weaker Students on **Class Test 3.**
Lecturer 29	Special Class for Weaker Students on **Class Test 3.**
Lecturer 30	Special Class for all Students on **Class Test 3.**
Lecturer 31	M2M to IoT-An Architectural Overview: Building architecture, Main design principles and needed capabilities.
Lecturer 32	M2M to IoT-An Architectural Overview: An IoT architecture outline, standards considerations.
Lecturer 33	IoT Reference Architecture: Introduction, Functional View, Information View.
Lecturer 34	IoT Reference Architecture: Deployment and Operational View, Other Relevant architectural views.
Lecturer 35	Revision Class from Lecturer 22 to Lecturer 24.
Lecturer 36	Revision Class from Lecturer 22 to Lecturer 24.
Lecturer 37	Revision Class from Lecturer 31 to Lecturer 34.
Lecturer 38	**Class Test** 4 on Lecturer 22 to Lecturer 24 and Lecturer 31 to Lecturer 34.
Lecturer 39	Review on **Class Test 4.**
Lecturer 40	Special Class for Weaker Students on **Class Test 4.**
Lecturer 41	Special Class for Weaker Students on **Class Test 4.**
Lecturer 42	Special Class for all Students on **Class Test 4.**
Lecturer 43	Real-World Design Constraints: Introduction, Technical Design constraints hardware is popular again.
Lecturer 44	Real-World Design Constraints: Data representation and visualization, Interaction and remote control.
Lecturer 45	Industrial Automation: Service-oriented architecture based device integration.
Lecturer 46	Revision Class from Lecturer 22 to Lecturer 24.
Lecturer 47	Revision Class from Lecturer 31 to Lecturer 34.
Lecturer 48	Revision Class from Lecturer 43 to Lecturer 45.
Lecturer 49	**Class Test** 5 on Lecturer 22 to Lecturer 24, Lecturer 31 to Lecturer 34 and Lecturer 43 to Lecturer 45.
Lecturer 50	Review on **Class Test 5.**
Lecturer 51	Special Class for Weaker Students on **Class Test 5.**
Lecturer 52	Special Class for Weaker Students on **Class Test 5.**
Lecturer 53	Special Class for all Students on **Class Test 5.**
Lecturer 54	SOCRADES: realizing the enterprise integrated Web of Things.
Lecturer 55	IMC-AESOP: from the Web of Things to the Cloud of Things.
Lecturer 56	**Class Test** 6 on Whole syllabus
Lecturer 57	Review on **Class Test 6.**
Lecturer 58	Special Class for Weaker Students on **Class Test 6.**
Lecturer 59	Special Class for Weaker Students on **Class Test 6.**
Lecturer 60	Special Class for all Students on **Class Test 6.**
Lecturer 61	Special Lecturer for all Students on their choice.
Lecturer 62	Special Lecturer for all Students on their choice.
Lecturer 63	Special Lecturer for all Students on their choice.
Lecturer 64	Special Lecturer for all Students on their choice.
Lecturer 65	Special Lecturer for all Students on their choice.
After completion of all lecturers Students will present a Seminar on this Course	

Course Outcome:	**After undergoing the course, Students will be able to:** Examine the potential business opportunities that IoT can uncover. Identify how IoT differs from traditional data collection systems. Use real IoT protocols for communication. Determine the right sensors and communication protocols to use in a particular IoT system. Establish data migration techniques from IoT devices to the cloud. Implement security features to protect data stored in the cloud. Understanding the fundamentals of Internet of things and Its architecture. Understand of IOT Protocols and IOT Applications.

Text/Reference Books:

1. Wireless Sensor Network by Sohraby, Minoli and Znati, Wiley Publications.
2. Wireless Sensor Network: A network perspective by Zheng & Abbas, Wiley.
3. Building Wireless Sensor Network by Faludi, O'Reilly.
4. Wireless Sensor Network: from theory to application by Ibrahiem, Ramakrishnan, CRC Press.
5. Wireless Sensor Network by H Mahmoud Ahmed Fahmy, Springer.
6. Internet of Things by Bahga, Madishetty, Orient Blackswanpvt Ltd.
7. IOT fundamentals, David, Pearson Education.
8. Internet of Things by Tripathy and Anuradha, CRC Press.
9. From Machine-to-Machine to the Internet of Things: Introduction to a New Age of Intelligence, Jan Holler, VlasiosTsiatsis, Catherine Mulligan, Stefan Avesand, Stamatis Karnouskos, David Boyle, Academic Press

Head
Dept. of Computer science & BCA
Gour Mahavidyalaya
Malda

GOUR MAHAVIDYALAYA
MANGALBARI, MALDA

LESSON PLAN

PROGRAM NAME: Computer Science (Honours)

SEM: 6th :: **PAPER: DC 13**

COURSE: SOFTWARE ENGINEERING

TEACHER NAME: Dr. Subhendu Chatterjee

DC13: Software Engineering: Introduction: Defining system, open and closed system, modeling of system through computer hardware, communication systems, external agents and software systems; Importance of Engineering Methodology towards computerization of a system Software Life Cycle: Classical and Iterative Waterfall Model; Spiral Model and its importance towards application for different system representations, Comparative Studies Software Requirement and Specification Analysis: Requirements Principles and its analysis principles; Specification Principles and its representations Software Design Analysis – Different level of DFD Design, Physical and Logical DFD, Use and Conversions between them, Process Representation – Pseudo English, Tight English, Decision Tables and Trees, Structured analysis – Structure Chart Conversion from DFD: Transform Centric and Transaction Centric conversions algorithms, Coupling and Cohesion of the different modules Software Cost Estimation Modeling – Heuristic and Empirical Modeling; COCOMO Software Testing: Software Verification and Validation; Testing objectives, Testing Principles, Testability; Error and Faults; Unit Testing, White Box and Blank Box Testing, Test Case Design: Test Vector. Software Quality Assurances: Concepts of Quality, Quality Control, Quality Assurance, SQA Activities, IEEE Standard for Statistical Software Quality Assurances (SSQA) criterions.

CLASS	TOPIC
Lecturer 1	Introduction: Defining system, open and closed system.
Lecturer 2	Modeling of system through computer hardware, communication systems, external agents and software systems.
Lecturer 3	Importance of Engineering Methodology towards computerization of a system Software.
Lecturer 4	Life Cycle: Classical and Iterative Waterfall Model.
Lecturer 5	Revise Lecturer 4
Lecturer 6	Life Cycle: Spiral Model and importance of all model towards application for different system representations.
Lecturer 7	Revise Lecturer 4 and Lecturer 6.
Lecturer 8	**Class Test 1** on Lecturer 1 to lecturer 7.
Lecturer 9	Review on **Class Test 1.**
Lecturer 10	Special Class for Weaker Students on **Class Test 1.**
Lecturer 11	Special Class for Weaker Students on **Class Test 1.**
Lecturer 12	Special Class for Weaker Students on **Class Test 1.**
Lecturer 13	Comparative Studies Software Requirement and Specification Analysis: Requirements Principles and its analysis principles.
Lecturer 14	Specification Principles and its representations Software Design Analysis – Different level of DFD Design.
Lecturer 15	Physical and Logical DFD, Use and Conversions between them.
Lecturer 16	Process Representation – Pseudo English, Tight English, Decision Tables and Trees, Structured analysis.
Lecturer 17	Revise Lecturer 13 to Lecturer 14.
Lecturer 18	Revise Lecturer 15 to Lecturer 16.
Lecturer 19	Revise Lecturer 1 to Lecturer 3 and Lecturer 6.
Lecturer 20	Revise Lecturer 4 and Lecturer 6.
Lecturer 21	**Class Test 2** on 1 to Lecturer 4 and Lecturer 6 as well as 13 to 16
Lecturer 22	Review on **Class Test 2.**
Lecturer 23	Special Class for Weaker Students on **Class Test 2.**

Lecturer 24	Special Class for Weaker Students on **Class Test 2.**
Lecturer 25	Special Class for Weaker Students on **Class Test 2.**
Lecturer 26	Structure Chart Conversion from DFD: Transform Centric and Transaction Centric conversions algorithms.
Lecturer 27	Coupling and Cohesion of the different modules Software Cost Estimation Modeling Heuristic and Empirical Modeling.
Lecturer 28	COCOMO Software Testing: Software Verification and Validation.
Lecturer 29	Revise Lecturer 26 to Lecturer 28.
Lecturer 30	Revise Lecturer 26 to Lecturer 28.
Lecturer 31	**Class Test 3** on Lecturer 26 to Lecturer 28
Lecturer 32	Review on **Class Test 3.**
Lecturer 33	Special Class for Weaker Students on **Class Test 3.**
Lecturer 34	Special Class for Weaker Students on **Class Test 3.**
Lecturer 35	Special Class for Weaker Students on **Class Test 3.**
Lecturer 36	Special lecturer for all students for their own choice.
Lecturer 37	Special lecturer for all students for their own choice.
Lecturer 38	Special lecturer for all students for their own choice.
Lecturer 39	Testing objectives, Testing Principles, Testability; Error and Faults.
Lecturer 40	Unit Testing, White Box and Blank Box Testing. Differentiate between White Box and Black Box testing.
Lecturer 41	**Class Test 4** on Lecturer 26 to Lecturer 28 and Lecturer 39 to Lecturer 40
Lecturer 42	Review on **Class Test 4.**
Lecturer 43	Special Class for Weaker Students on **Class Test 4.**
Lecturer 44	Special Class for Weaker Students on **Class Test 4.**
Lecturer 45	Special Class for Weaker Students on **Class Test 4.**
Lecturer 46	Test Case Design: Test Vector. Software Quality Assurances: Concepts of Quality.
Lecturer 47	Quality Control, Quality Assurance, SQA Activities.
Lecturer 48	IEEE Standard for Statistical Software Quality Assurances (SSQA) criterions.
Lecturer 49	**Class Test 5** on Lecturer 39 to Lecturer 40 and Lecturer 46 to Lecturer 48
Lecturer 50	Review on **Class Test 5.**
Lecturer 51	Special Class for Weaker Students on **Class Test 5.**
Lecturer 52	Special Class for Weaker Students on **Class Test 5.**
Lecturer 53	Special Class for Weaker Students on **Class Test 5.**
Lecturer 54	Special lecturer for all students for their own choice.
Lecturer 55	Special lecturer for all students for their own choice.
Lecturer 56	Special lecturer for all students for their own choice.
Lecturer 57	**Class Test 6** on Whole Syllabus
Lecturer 58	Review on **Class Test 5.**
Lecturer 59	Special Class for Weaker Students on **Class Test 6.**
Lecturer 60	Special Class for Weaker Students on **Class Test 6.**
Lecturer 61	Special Class for Weaker Students on **Class Test 6.**
Lecturer 62	Special lecturer for all students for their own choice on whole syllabus.
Lecturer 63	Special lecturer for all students for their own choice on whole syllabus.
Lecturer 64	Special lecturer for all students for their own choice on whole syllabus.
After completion of all lecturers Students will present a Seminar on this Course	
Course Outcome:	**After undergoing the course, Students will be able to:** Understand basic concepts of software engineering. Implement Software life cycle models and have a knowledge of different phases of Software life cycle. Calculate the cost & staff for a particular project at the start. Schedule their software in an appropriate way & make it track. Make an unambiguous SRS (software requirement specification) after collecting requirements of any client.

Text / Reference Books:

1. Software Engineering: A Practitioner's Approach by R.S. Pressman, McGraw-Hill.

2. An Integrated Approach to Software Engineering by P. Jalote, Narosa Publishing House.

3. Software Engineering by K.K. Aggarwal and Y. Singh, New Age International Publishers.

4. Software Engineering by I. Sommerville, Addison Wesle.

5. Software Engineering for Students by D. Bell, Addison-Wesley.

6. Fundamentals of Software Engineering by R. Mall, PHI.

Head
Dept. of Computer Science & BCA
Gour Mahavidyalaya
Malda

LESSON PLAN

PROGRAM NAME: Computer Science (Honours)

SEM: 6th :: **PAPER: DC 14**

COURSE: COMPILER DESIGN

TEACHER NAME: Akhil Kr. Das

DC14 : Compiler Design

Assemblers & Loaders, Linkers: One pass and two pass assembler, design of an assembler, Absolute loader, relocation and linking concepts, relocating loader and Dynamic Linking.
Introduction: Overview of compilation, Phases of a compiler
Lexical Analysis: Role of a Lexical analyzer, Specification and recognition of tokens, Symbol table, lex.
Parsing: Bottom up parsing- LR parser (SLR, LALR, CLR), yacc.
Intermediate representations: Three address code generation & representations , syntax directed translation, translation of types, control Statements
Storage organization: Activation records, stack allocation
Code Generation: Object code generation

Class	Topic
Lecture 1	Concept of System Programming ; Basic differences between application & system software
Lecture 2	Concept of Assemblers & Loaders, Linkers
Lecture 3	One pass and two pass assembler
Lecture 4	Differences between One pass and two pass assembler
Lecture 5	Revision on Lecturer 1 to Lecturer 4.
Lecture 6	**Class Test 1** on Lecturer 1 to Lecturer 4.
Lecture 7	Review on **Class Test 1.**
Lecture 8	Revision on **Class Test 1.**
Lecture 9	Revise Lecturer for weaker students on **Class Test 1.**
Lecture 10	Absolute loader
Lecture 11	Relocation and linking concepts
Lecture 12	Relocating loader
Lecture 13	Dynamic Linking
Lecture 14	Revision on Lecturer 10 to Lecturer 13.
Lecture 15	**Class Test 2** on Lecturer 10 to Lecturer 13.
Lecture 16	Review on **Class Test 2.**
Lecture 17	Revise Lecturer for weaker students on **Class Test 2.**
Lecture 18	Overview of compilation, Phases of a compiler
Lecture 19	Role of a Lexical analyzer
Lecture 20	Specification and recognition of tokens, Symbol table
Lecture 21	Concept and use of 'lex'
Lecture 23	Parsing: Bottom up parsing.

Lecture 24	LR parser (SLR, LALR, CLR)
Lecture 25	Concept and use of 'yacc'
Lecture 26	Revision on Lecturer 10 to Lecturer 13.
Lecture 27	Revision on Lecturer 10 to Lecturer 13.
Lecture 27	Revision on Lecturer 18 to Lecturer 25.
Lecture 28	Revision on Lecturer 18 to Lecturer 25.
Lecture 29	**Class Test 3** on Lecturer 10 to Lecturer 13 and Lecturer 18 to Lecturer 25.
Lecture 30	Review on **Class Test 3.**
Lecture 31	Revise Lecturer for weaker students on **Class Test 3.**
Lecture 32	Revise Lecturer for weaker students on **Class Test 3.**
Lecture 33	Three address code generation - concept
Lecture 34	Three address code generation & representations
Lecture 35	Syntax directed translation
Lecture 36	Translation of types
Lecture 37	Control Statements
Lecture 38	Revision on Lecturer 18 to Lecturer 25.
Lecture 39	Revision on Lecturer 18 to Lecturer 25.
Lecture 40	Revision on Lecturer 33 to Lecturer 37.
Lecture 41	Revision on Lecturer 33 to Lecturer 37.
Lecture 42	**Class Test 4** on Lecturer 18 to Lecturer 25 and Lecturer 33 to Lecturer 37.
Lecture 43	Review on **Class Test 4.**
Lecture 44	Revise Lecturer for weaker students on **Class Test 4.**
Lecture 45	Revise Lecturer for weaker students on **Class Test 4.**
Lecture 46	Storage organization - concept
Lecture 47	Activation records
Lecture 48	Stack allocation
Lecture 49	Revision on Lectures 46 - 48
Lecture 50	Revision on Lecturer 33 to Lecturer 37.
Lecture 51	Revision on Lecturer 33 to Lecturer 37.
Lecture 52	Revision on Lecturer 46 to Lecturer 50.
Lecture 53	**Class Test 5** on Lecturer 33 to Lecturer 37 and Lecturer 46 to Lecturer 50.
Lecture 54	Review on **Class Test 5.**
Lecture 55	Code Generation from the Intermediate code
Lecture 56	Object code generation – techniques and objectives
Lecture 57	Revision on Lectures 55 & 56
Lecture 58	**Class Test 6** on Whole syllabus

Lecture 59	Review on **Class Test 6.**
Lecture 60	Review of **Class Test 6** and special Lecturer for weaker students.
After completion of all lecturers Students will present a Seminar on this Course	
Course Outcome:	**After the completion of the course the student will be able to:** Understand how to build symbol tables and generate intermediate code. Understand compiler architecture. Design & conduct experiments for Intermediate Code Generation in compiler. Design & implement a software system for backend of the compiler. Deal with different translators.

Text/Reference Books:

1. Systems Programming, Santanu Chattopadhyaya, PHI.
2. Compilers: Principles, Techniques, and Tools, Alfred V. Aho, Monica S. Lam, Ravi Sethi, Jeffrey D. Ullman, Prentice Hall.
3. Systems Programming, D. M. Dhamdhere, Tata McGraw Hill.
4. System Software: An Introduction to System Programming, Leland Beck, D. Manjula, Pearson Education.
5. Modern Compiler Design, Grune D, Van Reeuwijk . K, Bal H. E, Jacobs C J H, Langendoen K, Springer

GOUR MAHAVIDYALAYA
MANGALBARI, MALDA

Head
Dept. of Computer science & BCA
Gour Mahavidyalaya
Malda

LESSON PLAN

PROGRAM NAME: Computer Science (Honours)

SEM: 6th :: **PAPER: DSE–3**

COURSE: INTRODUCTION TO DATA SCIENCE

TEACHER NAME: Arijit Bhattacharya & Ekram Alam

DSE-3:E2: Introduction to Data Science

Introduction: Data Analytics Lifecycle Overview, Data Preparation, Model Planning, Model Building; Clustering: K-means; Association Rules: Apriori Algorithm; Regression: Linear Regression; Classification: Decision Trees: Overview of a Decision Tree, Decision Tree Algorithms, Evaluating a Decision Tree; The Basics of NumPy Arrays; Basics of Data Manipulation with Pandas and Visualization with Matplotlib.

Class	Topic
Lecture 1	Introduction to Data Science
Lecture 2	Introduction to Data Science – continued
Lecture 3	Data Analytics Lifecycle Overview
Lecture 4	Data Analytics Lifecycle Overview – continued
Lecture 5	Revision on Lecturer 1 to Lecturer 4.
Lecture 6	**Class Test 1** on Lecturer 1 to Lecturer 4.
Lecture 7	Review on **Class Test 1.**
Lecture 8	Revision on **Class Test 1.**
Lecture 9	Revise Lecturer for weaker students on **Class Test 1.**
Lecture 10	Data Preparation – introduction
Lecture 11	Data Preparation – continue
Lecture 12	Model Planning
Lecture 13	Model Planning - continued
Lecture 14	Revision on Lecturer 10 to Lecturer 13.
Lecture 15	**Class Test 2** on Lecturer 10 to Lecturer 13.
Lecture 16	Review on **Class Test 2.**
Lecture 17	Revise Lecturer for weaker students on **Class Test 2.**
Lecture 18	Model Building – introduction
Lecture 19	Model Building – continued
Lecture 20	Clustering – concept
Lecture 21	Clustering – continued
Lecture 23	K-means; Association Rules – introduction
Lecture 24	K-means; Association Rules – continued
Lecture 25	K-means; Association Rules – continued
Lecture 26	Revision on Lecturer 10 to Lecturer 13.

Lecture 27	Revision on Lecturer 10 to Lecturer 13.
Lecture 27	Revision on Lecturer 18 to Lecturer 25.
Lecture 28	Revision on Lecturer 18 to Lecturer 25.
Lecture 29	**Class Test 3** on Lecturer 10 to Lecturer 13 and Lecturer 18 to Lecturer 25.
Lecture 30	Review on **Class Test 3.**
Lecture 31	Revise Lecturer for weaker students on **Class Test 3.**
Lecture 32	Revise Lecturer for weaker students on **Class Test 3.**
Lecture 33	Apriori Algorithm – introduction
Lecture 34	Apriori Algorithm – continued
Lecture 35	Regression – introduction
Lecture 36	Regression – continued
Lecture 37	Linear Regression
Lecture 38	Revision on Lecturer 18 to Lecturer 25.
Lecture 39	Revision on Lecturer 18 to Lecturer 25.
Lecture 40	Revision on Lecturer 33 to Lecturer 37.
Lecture 41	Revision on Lecturer 33 to Lecturer 37.
Lecture 42	**Class Test 4** on Lecturer 18 to Lecturer 25 and Lecturer 33 to Lecturer 37.
Lecture 43	Review on **Class Test 4.**
Lecture 44	Revise Lecturer for weaker students on **Class Test 4.**
Lecture 45	Revise Lecturer for weaker students on **Class Test 4.**
Lecture 46	Decision Trees
Lecture 47	Overview of a Decision Tree
Lecture 48	Decision Tree Algorithms
Lecture 49	Evaluating a Decision Tree
Lecture 50	Revision on Lecturer 33 to Lecturer 37.
Lecture 51	Revision on Lecturer 33 to Lecturer 37.
Lecture 52	Revision on Lecturer 46 to Lecturer 50.
Lecture 53	**Class Test 5** on Lecturer 33 to Lecturer 37 and Lecturer 46 to Lecturer 50.
Lecture 54	Review on **Class Test 5.**
Lecture 55	The Basics of NumPy Arrays
Lecture 56	Basics of Data Manipulation with Pandas
Lecture 57	Visualization with Matplotlib
Lecture 58	**Class Test 6** on Whole syllabus
Lecture 59	Review on **Class Test 6.**
Lecture 60	Review of **Class Test 6** and special Lecturer for weaker students.
After completion of all lecturers Students will present a Seminar on this Course	

Course Outcome:	**After undergoing the course, Students will be able to:** Summarize the basic concepts of data science and its importance. Analyze the data quantitatively or categorically, measure of averages, variability. Identify different trends in scatter plots, strengths of association between two numerical variables. Classify the concepts of data science and its importance. Build and assess data-based models. Execute statistical analyses with professional statistical software. Demonstrate skill in data management.

Text/Reference Books:

1. Introducing Data Science, D. Cielen, Arno D. B. Meysman, M. Ali, Dreamtech Press.
2. Doing Data Science: Straight Talk from the Frontiline , Rachel Schutt, Cathy O'Neil, by Schroff/O'Reilly.
3. Data Science for Business" What You Need to Know About Data Mining and Data-Analytic Thinking Foster Provost, Tom Fawcett by O'Reilly.
4. Data Smart: Using data Science to Transform Information into Insight, John W. Foreman, by John Wiley & Sons.
5. Super Crunchers: Why Thinking-by-Numbers Is the New Way to Be Smart, Ian Ayres by Bantam.
6. Python Data Science Handbook: Essential Tools for Working with Data, Jake VanderPlas by O'Reilly

GOUR MAHAVIDYALAYA
MANGALBARI, MALDA

Head
Dept. of Computer science & BCA
Gour Mahavidyalaya
Malda

LESSON PLAN

PROGRAM NAME: Computer Science (Honours)

SEM: 6th :: **PAPER: SEC–2**

COURSE: INTERNET TECHNOLOGY & WEB DESIGN

TEACHER NAME: Deb Pratim Sinha & Shib Charan Chowdhury

SEC–2 : Internet Technology & Web Design

The Internet and WWW: Understanding the WWW and the Internet, Emergence of Web, Web Servers, Web Browsers, Protocols. Building Web Sites, Planning for designing Web pages, Model and structure for a Website, Developing Websites, Basic HTML, Lists, Tables and Forms, Frames for designing a good interactive website.

Introduction to Web Design: Introduction to hypertext markup language (html) document type definition, creating web pages, graphical elements, lists, hyperlinks, tables, web forms, inserting images, frames. Introduction; Relative Links, Absolute Links; Link Attributes; Using the ID Attribute to Link Within a Document; Putting an Image on a Page, Using Images as Links, Putting an Image in the Background, Creating a Table, Table Headers, Captions, Spanning Multiple Columns, Styling Table.

PHP introduction, inventions and versions, scope, important tools and software requirements (like Web Server, Database, Editors etc.), Basic Syntax, PHP variables and constants, Types of data in PHP , Expressions, scopes of a variable (local, global), PHP Operators: Arithmetic, Assignment, Relational , Logical operators, Bitwise , ternary and MOD operator, PHP operator Precedence and associatively

Handling HTML form with PHP: Basic Input and Attributes, Other Kinds of Inputs, Styling forms with CSS, Where to Go from Here Capturing Form Data, GET and POST form methods, Dealing with multi value fields, Redirecting a form after submission

PHP conditional events and Loops: PHP IF Else conditional statements (Nested IF and Else), Switch case, while, For and Do While Loops, Goto, Break, Continue and exit.

PHP Functions: Function, Need of Function, declaration and calling of a function, PHP Function with arguments, Default Arguments in Function, Function argument with call by value, call by reference, Scope of Function Global and Local, Creating and accessing String, Searching & Replacing String; Formatting, joining and splitting String, String Related Library functions;

Array: Anatomy of an Array ,Creating index based and Associative array, Accessing array; Looping with Index based array, with associative array using each() and foreach(); Some useful Library function

Class	Topic
Lecture 1	Understanding the WWW and the Internet, Emergence of Web
Lecture 2	Web Servers, Web Browsers, Protocols
Lecture 3	Building Web Sites, Planning for designing Web pages, Model and structure for a Website
Lecture 4	Developing Websites, Basic HTML, Lists, Tables and Forms, Frames for designing a good interactive website
Lecture 5	Revision on Lecturer 1 to Lecturer 4.
Lecture 6	**Class Test 1** on Lecturer 1 to Lecturer 4.
Lecture 7	Review on **Class Test 1.**
Lecture 8	Revision on **Class Test 1.**
Lecture 9	Revise Lecturer for weaker students on **Class Test 1.**
Lecture 10	Introduction to hypertext markup language (html) document type definition, creating web pages,
Lecture 11	graphical elements, lists, hyperlinks, tables, web forms, inserting images, frames.
Lecture 12	Introduction; Relative Links, Absolute Links; Link Attributes; Using the ID Attribute to Link Within a Document; Putting an Image on a Page, Using Images as Links,

Lecture 13	Putting an Image in the Background, Creating a Table, Table Headers, Captions, Spanning Multiple Columns, Styling Table
Lecture 14	Revision on Lecturer 10 to Lecturer 13.
Lecture 15	**Class Test 2** on Lecturer 10 to Lecturer 13.
Lecture 16	Review on **Class Test 2.**
Lecture 17	Revise Lecturer for weaker students on **Class Test 2.**
Lecture 18	PHP introduction, inventions and versions, scope, important tools and software requirements (like Web Server, Database, Editors etc.)
Lecture 19	Basic Syntax, PHP variables and constants,
Lecture 20	Types of data in PHP
Lecture 21	Expressions, scopes of a variable (local, global)
Lecture 23	Arithmetic, Assignment, Relational
Lecture 24	Logical operators, Bitwise operators
Lecture 25	Ternary and MOD operator
Lecture 26	Revision on Lecturer 10 to Lecturer 13.
Lecture 27	Revision on Lecturer 10 to Lecturer 13.
Lecture 27	Revision on Lecturer 18 to Lecturer 25.
Lecture 28	Revision on Lecturer 18 to Lecturer 25.
Lecture 29	**Class Test 3** on Lecturer 10 to Lecturer 13 and Lecturer 18 to Lecturer 25.
Lecture 30	Review on **Class Test 3.**
Lecture 31	Revise Lecturer for weaker students on **Class Test 3.**
Lecture 32	Revise Lecturer for weaker students on **Class Test 3.**
Lecture 33	PHP operator Precedence and associatively
Lecture 34	Handling HTML form with PHP: Basic Input and Attributes
Lecture 35	Other Kinds of Inputs, Styling forms with CSS
Lecture 36	Where to Go from Here Capturing Form Data
Lecture 37	GET and POST form methods, Dealing with multi value fields Redirecting a form after submission
Lecture 38	Revision on Lecturer 18 to Lecturer 25.
Lecture 39	Revision on Lecturer 18 to Lecturer 25.
Lecture 40	Revision on Lecturer 33 to Lecturer 37.
Lecture 41	Revision on Lecturer 33 to Lecturer 37.
Lecture 42	**Class Test 4** on Lecturer 18 to Lecturer 25 and Lecturer 33 to Lecturer 37.
Lecture 43	Review on **Class Test 4.**
Lecture 44	Revise Lecturer for weaker students on **Class Test 4.**
Lecture 45	Revise Lecturer for weaker students on **Class Test 4.**
Lecture 46	**PHP Functions:** Function, Need of Function, declaration and calling of a function, PHP Function with arguments
Lecture 47	Default Arguments in Function, Function argument with call by value, call by reference, Scope of Function Global and Local

Lecture 48	Creating and accessing String, Searching & Replacing String
Lecture 49	String Formatting, joining and splitting String, String Related Library functions;
Lecture 50	Revision on Lecturer 33 to Lecturer 37.
Lecture 51	Revision on Lecturer 33 to Lecturer 37.
Lecture 52	Revision on Lecturer 46 to Lecturer 50.
Lecture 53	**Class Test 5** on Lecturer 33 to Lecturer 37 and Lecturer 46 to Lecturer 50.
Lecture 54	**Review on Class Test 5.**
Lecture 55	Anatomy of an Array , Creating index based and Associative array
Lecture 56	Accessing array; Looping with Index based array, with associative array using each() and foreach();
Lecture 57	Some useful Library function
Lecture 58	**Class Test 6** on Whole syllabus
Lecture 59	**Review on Class Test 6.**
Lecture 60	Review of **Class Test 6** and special Lecturer for weaker students.
After completion of all lecturers Students will present a Seminar on this Course	
Course Outcome:	**After undergoing the course, Students will be able to:** Gain an in-depth understanding of the web and Internet technologies. Understand the basic concepts for network implementation. Understand the basic working scheme of the Internet and the World Wide Web. Learn about the fundamental tools and technologies for web design. Comprehend the technologies for Hypertext Mark-up Language (HTML). Specify the design rules in constructing websites and web pages. Effectively deal with the programming issues related to HTML,CSS, and PHP.

Text/Reference Books:

1. Integrated HTML and CSS A Smarter, Virginia DeBolt, Faster Way to Learn, Wiley / Sybex.
2. Introduction to HTML and CSS, Cassidy Williams, Camryn Williams, O'Reilly.
3. XML in action web technology by William J. Pardi.
4. Step by Step XML by Michael J. Young.
5. PHP: The Complete Reference Paperback, Steven Holzner, McGraw Hill Education (India).
6. Timothy Boronczyk, Martin E. Psinas, "PHP and MYSQL (Create-Modify-Reuse)", Wiley India Private Limited.
7. Learning PHP, MySQL, JavaScript, CSS & HTML5, Robin Nixon, O'reilly.
8. PHP and MySQL Web Development, Luke Welling, Laura Thompson, Addison-Wesley Professional.
9. Web Enabled Commercial Application Development Using Html, Dhtml, Javascript, Perl Cgi By Ivan Bayross, BPB Publications.

D. Sinha

GOUR MAHAVIDYALAYA MANGALBARI, MALDA

Head
Dept. of Computer science & BCA
Gour Mahavidyalaya
Malda