ASSAM SCIENCE AND TECHNOLOGY UNIVERSITY (ASTU)

Course Structure

Computer Science & Engineering

Semester III:

Sl No	Sub- Code	Subject	Hrs			Credits
			L	T	P	C
	Theory					
1	MA131301	Mathematics III	3	2	0	4
2	MA131302	Discrete Mathematics	3	2	0	4
3	CS131303	Object Oriented Programming in C++	3	2	0	4
4	ET131304	Digital Systems	3	2	0	4
5	CS131305	Data Structure and Algorithms	3	2	0	4
6	HS131306	Sociology	2	0	0	2
	Practical					
7	CS131313	Object Oriented Programming in C++	0	0	2	1
8	ET131314	Digital Systems Lab	0	0	2	1
9	CS131315	Data Structure and Algorithms lab	0	0	2	1
		Total	17	10	6	25
Total Working Hours = 33						
Total Credits = 25						

Course Title: MATHEMATICS III
Course Code: MA131301
L-T-:: C **3-2=4**
Abstract:

This course of Mathematics is important for almost all the engineering disciplines. It deals with the partial differential equations of first order and 2nd order.

Prerequisites: Concept of solution of ODE, Elementary complex numbers and properties, Elementary probability and statistics – measures of central tendency, dispersions. Basic differentiation and integration [HS / diploma level]

Course Outcomes:

The students will

- ❖ Be able to apply the fundamental concepts of Partial differential Equations.
- ❖ Get familiarised with the applications of Ordinary Differential Equations and Partial Differential Equations.
- ❖ Be able to apply different techniques of integration, including partial fractions, integration by parts and recurrence formulae, to solve problems.

Module	Topic	No of hours	Marks
1	**First order Partial differential equation:** Partial differential equation of first order, Linear partial differential equation, Non-linear partial differential equation, Homogenous and non-homogeneous partial differential equation with constant co-efficient, Cauchy type, Monge's method. **Second order Partial differential equation:** Second order partial differential equation The vibrating string, the wave equation and its solution, the heat equation and its solution, Two dimensional wave equation and its solution, Laplace equation in polar, cylindrical and spherical coordinates, potential.	15	30
2	**Complex Analysis:** Analytic function, Cauchy-Riemann equations, Laplace equation, Conformal mapping, Complex integration: Line integral in the complex plane, Cauchy's integral theorem, Cauchy's integral formula, Derivatives of analytic functions. **Mathematical Series**: Power Series, Taylor's series, Laurent's series, Singularities and zeros, Residue integration method.	15	30
3	**Probability and statistics:** (i)Definition of probability, Laws of probability, Bays theorem, random variables, probability distributions and characteristics, binomial distribution, poisson's distributions and Normal distribution. (ii) elementary sampling theory, tests of hypothesis (statistical inference), Standard error, Fudicial (Confidence) limits, Tests of significance- Students' T-tests, Chi square tests and Z –tests.	10	25

4	**Laplace Transform** Definition of Laplace transform, Laplace transform of elementary functions, inverse of Laplace transforms. Properties of Laplace Transform- Linearity, multiplication by t^n and division by t. Laplace Transform of derivatives and integrals. Shifting theorems, Laplace transform of (i) periodic function (ii) unit step function, (iii) Dirac-delta function. Covolution theorem, Application of Laplace transform to initial value problems.	8	15

Reference books:

1. E. Kreyszig," Advanced Engineering Mathematics:, Eighth Edition, Wiley India.
2. B.V. Ramana, "Higher Engineering Mathematics", McGraw Hil Education.
3. N.P.Bali and Manish Goel, " A text book of Engineering mathematics", Laxmi Publication.
4. B. S. Grewal , "Higher Engineering Mathematics", Khanna Publication, Delhi.
5. Babu Ram, "Engineering Mathematics", Pearson.

SUBJECT: Discrete Mathematics
CODE: MA131302
L-T-C: 3-2--4
CLASS HOURS: 5 hrs./ Week
EXPECTED NO OF WEEKS: 9 (APPROX)
TOTAL NO OF CLASSES: 41 (APPROX)
L-Lectures, T-Tutorials, P-Practicals, C-Credits

COURSE CONTENTS:

	Hours	
1. MODULE I :	**4**	
Sets, countable/uncountable sets, integers, induction		
2. MODULE II:	**4**	
Functions, relations, equivalence classes, partitions.		
3. MODULE III :	**9**	
Propositional logic, Boolean algebra		
4. MODULE IV :	**9**	

Abstract Algebra – Basics of groups, rings, finite fields, vector spaces.		
5. MODULE V :	**6**	
Combinatorics – Counting principles, recurrence equations, generating functions.		
6. MODULE VI :		
Probability Theory – Sample space, events, expectations, variance, distribution, random variables, binomial, poisson and geometric random variables	**9**	

TEXT BOOKS:

1. "Discrete Mathematical Structures with Applications to Computer Science", by J.P. Tremblay and R. Monohar.

2. "Discrete Mathematics and Its Applications", by Kenneth H. Rosen, Tata McGraw Hill, 6th edition, ISBN: 0072880082© 2007

3. " Elements of Discrete Mathematics", by C. L. Liu, Tata McGraw Hill Education Private Limited, 3rd edition,2008

SUBJECT: Object Oriented Computer Programming in C++
CODE: CS131303
L-T-C: 3-2-4
CLASS HOURS: 4 hrs./ Week
EXPECTED NO OF WEEKS: 13 (APPROX)
TOTAL NO OF CLASSES: 39 (APPROX)
L-Lectures, T-Tutorials, C-Credits

OBJECTIVES:

1. Acquire an understanding of basic object-oriented concepts and the issues involved in effective class design.
2. Write C++ programs that use: object-oriented concepts such as information hiding, constructors, destructors, inheritance

PREREQUISITE:

1. Introduction to Computer Programming (CS131105)

FOR TEACHERS: **COURSE CONTENTS:**	**Hours**	
7. MODULE I : INTRODUCTION		
(a) What is Object Oriented Programming? Why we need Object Oriented Programming? Programming characteristics of OOP. Difference between OOP and procedure oriented programming;	**2**	
(b) Basic Concepts of OOPs, feature of OOPs, Application of OOPs, and .Review of Data Types (user define and derived data types), Keywords, Tokens, Identifies, Constants, Reference variables, different Operators and Control statements	**5**	
8. MODULE II: CLASSES AND OBJECTS		
(a) Introduction to Objects and classes, Difference between Class and Structure, Class definition and syntax, Defining member functions, Access control to other functions(Private, Public, Protected)	**2**	
(b) Objects-Dynamic Creation and initialization, Passing and Returning objects, Object assignment and array of objects,;	**2**	
(c) Constructors-Types, Destructors, Nesting member function, Private member function , Inline functions,	**2**	
(d) Static class members, Function prototyping, Call by reference, Return by reference, Default Argument, Friend functions, this pointer.	**2**	
9. MODULE III : INHERITANCE		
(a) Types of Inheritance; Base and Derived classes – Syntax of derived classes, access to the base class; Types of Inheritance,	**2**	
(b) Multiple inheritance – Virtual Base classes, Constructors and Destructors in Inheritance,	**3**	
(c) Container classes, Abstract Classes.	**1**	
10. MODULE IV : POLYMORPHISM		
a) Compile time(Early/Static binding)-Overloading functions and operators, Overloading new and delete operators;	**2**	
b) Run time polymorphism(Late/Dynamic Binding) – Virtual functions, Pure Virtual functions, Virtual Destructors,	**3**	
c) Review of Virtual base classes,	**1**	
11. MODULE V : TEMPLATES		
Templates – Uses, Generic classes, Class templates, Function templates, Advance templates. Examples	**2**	

12. MODULE VI : EXCEPTION HANDLING AND MANIPULATORS		
(a) Exception handling – Advantages, Try catch and throw clauses, Examples, (b) Manipulators, different examples of manipulators;	**2** **2**	
13. MODULE VII: POINTERS AND FILES		
a) Pointer types- uses; Dynamic memory allocation techniques - garbage collection, Linked list, generic pointers; b) Files- Open, Close, Read and Write; File attributes, File management	**3** **3**	

TEXT/ REFERENCE BOOKS:

1. E. BALAGURUSWAMY**: OBJECT ORIENTED PROGRAMMING WITH C++** , Tata McGraw Hill.
2. HERBERT SCHILDT: "**C++, THE COMPLETE REFERENCE"**
3. BARKATAKI : "**OBJECT ORIENTED PROGRAMMING"**, PHI

REFERENCES:

1. DEITAL AND DEITAL **: "C++ HOW TO PROGRAM"**
2. O'REILY: **"Head First C#:"**
3. R. LAFORE : "**OBJECT ORIENTED PROGRAMMING IN TURBO C++",** GALGOTIA, NEW DELHI
4. P.B. MAHAPATRA : "**THINKING IN C- INCLUDING OBJECT ORIENTED PROGRAMMING WITH C++"** , WHEELER PUBLISHING

SUBJECT: DIGITAL SYSTEMS	
CODE:	ET131304
L-T--C:	**3-2-4**
CLASS HOURS	**5 hrs./ Week**
EXPECTED NO OF WEEKS:	**10 (APPROX)**
TOTAL NO OF CLASSES:	**48 (APPROX)**
L-Lectures, T-Tutorials, C-Credits	
PREREQUISITES: I. Basic Electrical & Electronics Engineering-I(EE131104) II. Basic Electrical & Electronics Engineering-II(ET131204)	
OBJECTIVES:	

➢ **Understanding of Simplification of boolean expression and how to implement with various gates.** ➢ **Concepts of Combinational and Sequential Circuits design.** ➢ **Knowledge on digital logic families.**		
FOR TEACHERS:		
COURSE CONTENTS:		
	Hours	**Max Marks**
14. MODULE I : Fundamental Of Digital Electronics	**8**	**20**
Review of number system; Position number system – decimal, binary, octal and hexadecimal, number base conversion .Representation of negative binary numbers. Codes – BCD Gray , Excess -3 Digital signal , logic gates : AND, OR, NOT, NOR, EX-OR, EX-NOR		
15. MODULE II : Boolean algebra and its simplification	**12**	**20**
Axioms and basic theorem of Boolean algebra. Truth table , logic function and their realization , standard representation (canonical forms) of logic gates.-SOP and POS forms, MIN terms and MAX terms Simplification of logic function: K-map of 2, 3, 4 and 5 variables. Simplification of algebra and by map method. Don't care condition. Quine Mcluskey methods of simplification. Synthesis using AND, OR and INVERT and then to convert to NAND or NOR implementation		
16. MODULE III : Combinational logic circuit design	**8**	**20**
Combinational logic circuit and buildings blocks. Binary adders and subs tractors. Carry look ahead addrer. Encoders, Decoders , Multiplexers , Demultiplexers , Comparators, parity generators, etc. Realization of logic gates functions through decoders and multiplexers.		
17. MODULE IV: Sequential circuits	**12**	**30**

Flip flops: truth table and state table SR, JK, TD, race around condition, master slave conversion of flip-flops. Sequential shift register, sequence generator. Counter s: asynchronous and d Synchronous generators. Ring counter s and Johnson counter, up. Down counter modulo – N counter. Design of Synchronous sequential circuit.		
18. **MODULE V: Digital logic families and programmable logic devices.**	**8**	**10**
Switching mode operation of PN junction , Bipolar and MOD device Bipolar families: RTL,DTL,DCTL,HTL,TTL,ECL,MOS, and CMOS logic families, Tristate logic. Gate properties fan in, fan out, propagation delay and power delay product. RAM and ROM their uses, SSI, MSI LSI and V LSI devices. Introduction to PLA, PAL TO FPGA and CPLDS, Some commonly used digital ICs		
TEXT/ REFERENCE BOOKS: 1. M. Morris Mano — Digital Design. Prentice Hall of India 2. P. Malvino an4 D. K. Leach— Digital Principles and Applications. TataMcGraw-Hill. 3. M. Morris Mano — Digital Logic and Computer Design. Prentice Hall of India.		

SUBJECT: Data Structure and Algorithm	
CODE:	**CS131305**
L-T-C:	**3-2-4**
CLASS HOURS	**5 hrs./ Week**
EXPECTED NO OF WEEKS:	**10 (APPROX)**
TOTAL NO OF CLASSES:	**46 (APPROX)**
L-Lectures, T-Tutorials, C-Credits	
PREREQUISITES:	

I. Introduction to Computing(CS131105), II. Mathematics-I(MA131101), Mathematics-II(MA131201),		
OBJECTIVES: • Understanding the data structures, their advantages and drawbacks, how to implement them in C & how they can be overcome. • Understanding their applications and their uses. • Students will learn about the data structure methods or algorithms mentioned in the course so as to make use of them in a program to enhance their efficiency (i.e. reduce the run-time) or for better memory utilization.		
FOR TEACHERS:		
COURSE CONTENTS:		
	Hours	**Max Marks**
19. MODULE I : LINEAR DATA STRUCTURE- I		
a. **Introduction:** Why we need data structure? Concepts of data structures: Data and data structure, Abstract Data Type and Data Type. Algorithms and programs, basic idea of pseudo-code. Algorithm efficiency and analysis, time and space analysis of algorithms – order notations.	**8**	**30**
b. Array: Different representations – row major, column major. Sparse matrix - its implementation and usage. Array representation of polynomials.	**2**	**20**
c) Linked List: Singly linked list, circular linked list, doubly linked list, linked list representation of polynomial and applications.	**4**	**20**
20. MODULE II : LINEAR DATA STRUCTURE- II		
a) Stack and Queue: Stack and its implementations (using array, using linked list), applications. Queue, circular queue, dequeue. Implementation of queue- both linear and circular (using array, using linked list), applications.	**5**	**20**
b) Recursion: Principles of recursion – use of stack, differences between recursion and iteration, tail recursion.	**2**	**10**

Applications - The Tower of Hanoi, Eight Queens Puzzle.		
21. MODULE III : NONLINEAR DATA STRUCTURES		
a) Trees: Basic terminologies, forest, tree representation (using array, using linked list). Binary trees - binary tree traversal (pre-, in-, post- order), threaded binary tree (left, right, full) - non-recursive traversal algorithms using threaded binary tree, expression tree. Binary search tree- operations (creation, insertion, deletion, searching). Height balanced binary tree – AVL tree (insertion, deletion with examples only). B- Trees – operations (insertion, deletion with examples only).	**9**	**30**
b) Graphs: Graph definitions and concepts (directed/undirected graph, weighted/un-weighted edges, sub-graph, degree, cut-vertex/articulation point, pendant node, clique, complete graph, connected components – strongly connected component, weakly connected component, path, shortest path, isomorphism). Graph representations/storage implementations – adjacency matrix, adjacency list, adjacency multi-list. Graph traversal and connectivity – Depth-first search (DFS), Breadth-first search (BFS) – concepts of edges used in DFS and BFS (tree-edge, back-edge, cross-edge, forward-edge), applications. Minimal spanning tree – Prim's algorithm (basic idea of greedy methods).	**6**	**20**
22. MODULE IV: SEARCHING, SORTING		
a) Sorting Algorithms: Bubble sort and its optimizations, insertion sort, shell sort, selection sort, merge sort, quick sort, heap sort (concept of max heap, application – priority queue), radix sort.	**6**	**10**
b) Searching: Sequential search, binary search, interpolation search.	**4**	**10**

TEXT/ REFERENCE BOOKS:

1. **"Data Structures And Program Design In C", 2/E by Robert L. Kruse, Bruce P. Leung.**

2. **"Fundamentals of Data Structures of C" by Ellis Horowitz, Sartaj Sahni, Susan Anderson-freed.**
3. **"Data Structures in C" by Aaron M. Tenenbaum.**
4. **"Data Structures" by S. Lipschutz.**
5. **"Data Structures Using C" by Reema Thareja.**
6. **"Data Structure Using C", 2/e by A.K. Rath, A. K. Jagadev.**
7. **"Introduction to Algorithms" by Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein.**

HS131306	**SOCIOLOGY**	L = 2 T = 0 C = 2
Module-I	Sociology in the Industrial Perspective: Concept of sociology/ Sociology as a science?/ Sociology of work and industry/ Perspectives for sociological analysis of work/ Class conflict in industry/ Social impact of industrialization	12 Hours
Module-II	Work and Social Change: Nature of modern societies/ Emergence of industrial capitalism/ Technology and social change/ The information society after the industrial society/ Postmodernity/ Globalization and convergence/ Significance of the service sector today/ Work restructuring and corporate management	12 Hours
Module-III	Work Experiences in Industry: The concept of alienation/ Work satisfaction/ Technology and work experience/ Social background of workers/ Work orientations/ Stress and anxiety of the worker/ Work and leisure/ Unemployment/ Conflicts in the workplace	12 Hours
	Total	36 Hours

Reference Books

1. Miller and Form, Industrial Sociology (London: Harper & Row, 1968)
2. N. R. Sheth, Social Framework of Indian Factory (Bombay: Oxford University Press, 1968)
3. Gisbert, Fundamentals of Industrial Sociology (New Delhi: Oxford University Press, 1971)
4. P. Gisbert, Fundamentals of Industrial Sociology (New Delhi: Oxford University Press, 1971)
5. Tony J. Watson, Sociology, Work and Industry (New York: Routledge, 2004 reprint)

SUBJECT: Object Oriented Computer Programming in C++ Lab
CODE: CS131313
L-T-P-C: 0-0-2-1
CLASS HOURS: 2 hrs./ Week
EXPECTED NO OF WEEKS: 12 (APPROX)
TOTAL NO OF CLASSES: 9 (APPROX)
L-Lectures, T-Tutorials, P-Practicals, C-Credits

OBJECTIVES:

1. To make the student to learn C++ programming language.

2. To teach the student the implementation of object oriented programming features.
3. To teach the student to write programs in C++ to solve the problems

PREREQUISITE:

1. Introduction to Computer Programming (CS131105)
2. Object Oriented computer Programming in C++ (CS131303)

LIST OF PROGRAMS:

Topics should include but not limited to:

23. MODULE I : INTRODUCTION	
[1] Write a C++ program to display "HELLO WORLD". [2] Write a C++ program that will ask the temperature in Fahrenheit and display in Celsius [3] Write a C++ program to print the following output using *for* loop. 1 2 2 3 3 3 4 4 4 4 [4] Write a C++ program to reverse a number using *do-while* loop [5] Write a C++ program to find out the factorial of a number using *while* loop [6] Write a C++ program to read an integer array and display it. [7] Write a C++ program to read a character array and display it. [8] Write a C++ program to find out the maximum of three number using *if-else* statement	**[8 programs form this module]**
24. MODULE II: CLASSES AND OBJECTS	
(e) Write a C++ program to implement the concept of static data member in class. **(f)** Write a C++ program to implement the concept of static function in class. **(g)** Write a C++ program using function with default argument. **(h)** Write a C++ program to illustrate the use of objects as function arguments (which performs the addition of time in the hour and minutes format)	**[8 programs form this module]**

(i) Write a C++ program to illustrate the use of friend function. **(j)** Write a C++ program to illustrate how an object can be created (within a function) and returned to another function **(k)** Write a C++ program to illustrate the use of constructors and destructors. **(l)** Write a C++ program to illustrate the use of copy constructor.	
25. MODULE III : INHERITANCE	
[1] Write a C++ program to implement single inheritance (private/public) [2] Write a C++ program to implement multilevel inheritance [3] Write a C++ program to implement multiple inheritances. [4] Write a C++ program to illustrate the use of virtual base class.	**[4 programs form this module]**
26. MODULE IV : POLYMORPHISM	
[1] Write a C++ program to overload unary minus operator [2] Write a C++ program to overload binary '+' operator [3] Write a C++ program to illustrate how an operator can be overloaded using friend function. [4] Write a C++ program to illustrate the use of run time polymorphism.	**[4 programs form this module]**
27. MODULE V : TEMPLATES	
[1] Write a C++ program to swap two variable using *function template*	**[1 programs form this module]**
28. MODULE VI : EXCEPTION HANDLING AND MANIPULATORS	
[1] Write a C++ program to implement *try(), catch(), throw()* function.	**[1 programs form this module]**
29. MODULE VII: POINTERS AND FILES	
[2] Write a C++ program to implement this pointer [3] Write a C++ program to illustrate the use of pointers to derived objects [4] Write a C++ program to illustrate the use of virtual function [5] Write a C++ program to open and close a file using *open(), close()* function [6] Write a C++ program to illustrate the use of *read(), write()* function	**[6 programs form this module]**

TEXT/ REFERENCE BOOKS:

4. E. BALAGURUSWAMY**: OBJECT ORIENTED PROGRAMMING WITH C++** , Tata McGraw Hill.
5. HERBERT SCHILDT: "**C++, THE COMPLETE REFERENCE"**
6. BARKATAKI : "**OBJECT ORIENTED PROGRAMMING"**, PHI

REFERENCES:

5. DEITAL AND DEITAL **: "C++ HOW TO PROGRAM"**
6. R. LAFORE : "**OBJECT ORIENTED PROGRAMMING IN TURBO C++",** GALGOTIA, NEW DELHI
7. P.B. MAHAPATRA : "**THINKING IN C- INCLUDING OBJECT ORIENTED PROGRAMMING WITH C++"** , WHEELER PUBLISHING

SUBJECT: Digital Systems Lab	
CODE:	ET131314
L-T-P-C:	**0-0-2-1**
CLASS HOURS	**2 hrs./ Week**
EXPECTED NO OF WEEKS:	**12(APPROX)**
TOTAL NO OF CLASSES:	**9 (APPROX)**
L-Lectures, T-Tutorials, P-Practicals, C-Credits	
PREREQUISITES:	
OBJECTIVES:	
FOR TEACHERS:	
COURSE CONTENTS:	
Experiments should include but not limited to :	
1. To study the TTL and CMOS families of Digital Integrated Circuits.	
2. To implement a simple Boolean expression on TTL/CMOS Small Scale Integrated Circuit (SSI) Devices.	
3. To implement Half adder & Full adder.	

4. To study Parallel Binary adder.	
5. To study a BCD to 7 Segment LED display decoder as an example of a multiple input and multiple output combinational digital circuit.	
6. To study the IC 741 & implement a function using IC 74151	
7. To study the J-K,D and T flip flops.	
8. To study a simple two-bit ripple counter.	
TEXT/ REFERENCE BOOKS:	

SUBJECT: Data Structure and Algorithm Lab	
CODE:	CS**131315**
L-T-P-C:	**0-0-2-1**
CLASS HOURS	**2 hrs./ Week**
EXPECTED NO OF WEEKS:	**12(APPROX)**
TOTAL NO OF CLASSES:	**9 (APPROX)**
L-Lectures, T-Tutorials, P-Practicals, C-Credits	
PREREQUISITES: III. Introduction to Computing(CS131105), IV. Mathematics-I(MA131101), Mathematics-II(MA131201),	
OBJECTIVES: • Understanding the data structures, their advantages and drawbacks, how to implement them in C & how they can be overcome. • Understanding their applications and their uses. • Students will learn about the data structure methods or algorithms mentioned in the course so as to make use of them in a program to enhance their efficiency (i.e. reduce the run-time) or for better memory utilization.	
FOR TEACHERS:	
COURSE CONTENTS:	
Experiments should include but not limited to :	
9. Implementation of array operations.	
10. Stacks and Queues: adding, deleting elements Circular Queue: Adding & deleting elements Merging Problem.	
11. Evaluation of expressions operations on Multiple stacks & queues.	
12. Implementation of linked lists: inserting, deleting, and inverting a linked list. Implementation of stacks & queues using linked lists	
13. Polynomial addition, Polynomial multiplication.	

14. Sparse Matrices : Multiplication, addition.	
15. Recursive and Nonrecursive traversal of Trees.	
16. Threaded binary tree traversal. AVL tree implementation.	
17. Application of Trees. Application of sorting and searching algorithms.	

TEXT/ REFERENCE BOOKS:

8. **"Data Structures And Program Design In C", 2/E by Robert L. Kruse, Bruce P. Leung.**
9. **"Fundamentals of Data Structures of C" by Ellis Horowitz, Sartaj Sahni, Susan Anderson-freed.**
10. **"Data Structures in C" by Aaron M. Tenenbaum.**
11. **"Data Structures" by S. Lipschutz.**
12. **"Data Structures Using C" by Reema Thareja.**
13. **"Data Structure Using C", 2/e by A.K. Rath, A. K. Jagadev.**
14. **"Introduction to Algorithms" by Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein.**

****