Assam University, Silchar

DEPARTMENT OF COMPUTER SCIENCE

Four Year Undergraduate (FYUG)/M. Sc. (Integrated) 5-year Programme

Under NEP 2020

Effective from the Academic Year 2023-24

Approved in the 94th meeting of the Academic Council on 20th July 2023
vide Resolution No AC:94:07-23:6

Head
Department of Computer Science
Assam University, Silchar
PIN 788011

Department of Computer Science
Assam University, Silchar

Curriculum for Four Year Undergraduate (FYUG)/M. Sc. (Integrated) 5-year Programme in Computer Science under NEP 2020 w.e.f. 2023-24

The Department of Computer Science offers B. Sc. (Computer Science) (Honours with Research), a Four Year Undergraduate Degree Programme (FYUG), M. Sc. (Integrated) 5-year programme and M. Sc. Degrees in Computer Science under the provisions of NEP 2020 effective from the Academic Year 2023-24.

Eligibility Criteria for admission to Four Year Undergraduate Degree programme/M. Sc. (Integrated) 5-year Programme in Computer Science:
Candidates who have passed the HS examination (10+2) in Science stream with Physics and Mathematics (in combination with any other science subjects) with at least 50% marks in aggregate are eligible for admission into the first semester of M. Sc. (Integrated) 5-year programme. Admission will be based on the scores of CUET conducted by National Testing Agency (NTA), Govt. of India.

Eligibility for promotion of students with three years undergraduate degree to the 7th Semester of the M. Sc. (Integrated) 5-year Programme (That is in the first semester of the PG level of the 5 year M. Sc. Integrated Programme), a student has to pass in all the papers up to the 6th semester of the Four Year Undergraduate Degree Programme and has to successfully complete the undergraduate degree of three years duration.

Eligibility for promotion of students with four years undergraduate degree to the 9th Semester of the M. Sc. (Integrated) 5-year Programme (That is in the third semester of the PG level of the M. Sc. (Integrated) 5-year Programme), a student has to pass in all the papers up to the 8th semester of the Four Year Undergraduate Degree Programme and has to successfully complete the undergraduate degree (Honours with Research) under the provisions of NEP 2020.

Course Mapping for CUET: 1. English 2. Three domain specific subjects viz; Physics and Mathematics as compulsory subjects and any one other science subject.

Selection of candidates for admission to the Four Year Undergraduate (FYUG)/M.Sc. (Integrated) 5-year Programme in Computer Science will be made on the basis of NTA-CUET score only.

Programme Specific Outcome

On successful completion of the FYUG Computer Science programme, students will be able to:

1. Understand and apply the concepts of computer architecture, data structure, networking and operating system.
2. Write and execute programs in various programming languages to solve real life problems.
3. Learn and apply techniques of database management systems and design data bases for various applications.
4. Understand the theories and algorithms of Artificial Intelligence including Machine Learning and Data Science and apply these to solve problems.
5. Perform analysis of algorithms related to space and time complexity of Algorithms.
6. Learn and apply the knowledge of Internet Technologies and Cyber Security.
7. Develop theoretical as well as practical knowledge of Software development.
8. Seek admission in Post-graduate/Research Programmes in Computer Science, Information Technology and Computer Application etc.
9. Seek employment in various jobs in the Government sector as well as IT and related industries and perform various roles related to software development, testing and maintenance.

SEMESTER-I

Course Code	Type of Course	Course Name	L	T	P	Credit	Sessional Marks	End Semester Marks	Total Marks
CS-DSC-101	Discipline Specific Core	Programming in C	3	-	-	3	30	70	100
CS-DSC-102	Discipline Specific Core	Digital Logic and Switching Theory	3	-	-	3	30	70	100
CS-DSM-101	Discipline Specific Minor	Mathematics-I	3	-	-	3	30	70	100
CS-IDC-101	Interdisciplinary Course	Fundamentals of Computer and Applications	3	-	-	3	30	70	100
CS-AEC-101 MIL-101	Ability Enhancement Courses (Language)	MIL-I	2	-	-	2	15	35	50
CS-SEC-101	Skill Enhancement Course	Lab on a) Programming in C b) Digital logic and Switching Theory	-	-	6	3	30	70	100
CS-VAC-101	Common Value Added Courses	NCC/NSS/Sports/ HW/UI/GCS/Yoga	3				30	70	100
SEMESTER I Total Marks: 650									

SEMESTER-II

Course Code	Type of Course	Course Name	L	T	P	C	Sessional Marks	End Semester Marks	Total Marks
CS-DSC-151	Discipline Specific Core	Python Programming	3	-	-	3	30	70	100
CS-DSC-152	Discipline Specific Core	Numerical Methods	3	-	-	3	30	70	100
CS-DSM-151	Discipline Specific Minor	Mathematics-II	3	-	-	3	30	70	100
CS-IDC-151	Interdisciplinary Course	Introduction to Internet Technology	3	-	-	3	30	70	100
CS-AEC-151 EL-151	Ability Enhancement Courses (Language)	English-I	2	-	-	2	15	35	50
CS-SEC-151	Skill Enhancement Course	Lab on a) Python Programming b) Numerical Methods	-	-	6	3	30	70	100
CS-VAC-151	Common Value Added Courses	EVS					30	70	100

Undergraduate Certificate: 40 Credits **SEMESTER II Total Marks: 650**

SEMESTER-III

Course Code	Type of Course	Course Name	L	T	P	C	Sessional Marks	End Semester Marks	Total Marks
CS-DSC-201	Discipline Specific Core	Data Structure	3	1	-	4	30	70	100
CS-DSC-202	Discipline Specific Core	Computer Architecture	3	1	-	4	30	70	100
CS-DSM-201	Discipline Specific Minor	Introduction to Probability and Statistics	3	1	-	4	30	70	100
CS-IDC-201	Interdisciplinary Course	Cyber Security	3	-	-	3	30	70	100
CS-AEC-201 MIL-201	Ability Enhancement Courses (Language)	MIL-II	-	-	-	2	15	35	50
CS-SEC-201	Skill Enhancement Course	Lab on Data Structure	-	-	6	3	30	70	100
SEMESTER III Total Marks: 550									

SEMESTER IV

Course Code	Type of Course	Course Name	L	T	P	C	Sessional Marks	End Semester Marks	Total Marks
CS-DSC-251	Discipline Specific Core	Database Management System	3	1	-	4	30	70	100
CS-DSC-252	Discipline Specific Core	Microprocessor	3	1	-	4	30	70	100
CS-DSC-253	Discipline Specific Core	Discrete Mathematics	3	1	-	4	30	70	100
CS-DSM-251	Discipline Specific Minor	Lab on a) Database Management System b) Microprocessor	-	-	6	3	30	70	100
CS-DSM-252	Discipline Specific Minor	Data Communication and Computer Networks	-	-	-	3	30	70	100
CS-AEC-251 EL-251	Ability Enhancement Courses (Language)	English-II	-	-	-	2	15	35	50
Undergraduate Diploma: 80 Credits								**SEMESTER IV Total Marks: 550**	

SEMESTER V

Course Code	Type of Course	Course Name	L	T	P	C	Sessional Marks	End Semester Marks	Total Marks
CS-DSC-301	Discipline Specific Core	Operating System	3	1	-	4	30	70	100
CS-DSC-302	Discipline Specific Core	System Software	3	1	-	4	30	70	100
CS-DSC-303	Discipline Specific Core	Computer Graphics	3	1	-	4	30	70	100
CS-DSM-301	Discipline Specific Minor	Lab on a) Operating System b) Computer Graphics	-	-	6	3	30	70	100
CS-DSM-302	Discipline Specific Minor	Simulation and Modeling	-	-	6	3	30	70	100
CS-SEC-301		Internship/ Community Engagement/ Field Study				2	15	35	50
SEMESTER V Total Marks: 550									

SEMESTER VI

Course Code	Type of Course	Course Name	L	T	P	C	Sessional Marks	End Semester Marks	Total Marks
CS-DSC-351	Discipline Specific Core	Object Oriented Programming with C++	3	1	-	4	30	70	100
CS-DSC-352	Discipline Specific Core	Programming in JAVA	3	1	-	4	30	70	100
CS-DSC-353	Discipline Specific Core	Wireless and Mobile Computing	3	1	-	4	30	70	100
CS-DSC-354	Discipline Specific Core	System Analysis and Design	3	1	-	4	30	70	100
CS-DSM-351	Discipline Specific Minor	Lab on a) Object Oriented Programming with C++ b) Programming in JAVA	-	-	8	4	30	70	100
Bachelor's Degree: 120 credits		**Min 7.5 CGPA to Move to 4th year**					**SEMESTER VI Total Marks: 500**		

SEMESTER VII

Course Code	Type of Course	Course Name	L	T	P	C	Sessional Marks	End Semester Marks	Total Marks
CS-DSC-401	Discipline Specific Core	Design and Analysis of Computer Algorithms	3	1	-	4	30	70	100
CS-DSC-402	Discipline Specific Core	Theory of Computation	3	1	-	4	30	70	100
CS-DSC-403	Discipline Specific Core	Artificial Intelligence	3	1	-	4	30	70	100
CS-DSC-404	Discipline Specific Core	Machine Learning	3	1	-	4	30	70	100
CS-DSM-401	Discipline Specific Minor	Lab on a) Design and Analysis of Computer Algorithms b) Artificial Intelligence	-	-	8	4	30	70	100
SEMESTER-VII Total Marks: 500									

SEMESTER VIII

Course Code	Type of Course	Course Name	L	T	P	C	Sessional Marks	End Semester Marks	Total Marks
CS-DSC-451	Discipline Specific Core	Principles of Compiler Design	3	1	-	4	30	70	100
CS-DSM-451	Discipline Specific Minor	Data Science and Research Methodology	3	1	-	4	30	70	100
CS-DSC-452		Research Project/ Dissertation	-	-	24	12	90	210	300
Bachelor's Degree (Honours with Research): 160 Credits								**SEMESTER VIII Total Mark: 500**	

Semester I

Course Code: CS-DSC-101 **Course Title: Programming in C**

Credits: 3 **LTP: 3 – 0 – 0**

Course objectives:
1. To introduce the concepts of programming and programming language C.
2. To explain the concepts of functions and programme structure in C
3. To explain how to write and implement C programs
4. To explain the concept and working of pointers and files in C
5. To introduce the low level programming in C

***Course outcomes*:**
After successful completion of the course, the students will be able to:
1. Apply the concepts of Programming in C
2. Apply thoroughly the building blocks of C programming language
3. Apply and implement C programs and solve problems through programming
4. Apply the Concept of pointers and files in C & Programming with C.
5. Design and implement programs using pointers and files in C.

UNIT-I:
Fundamentals of computer programming with C – Data Types, Expressions, Operations – input, output; Writing simple C programs; Control structures (WHILE, DO-WHILE, FOR, IF-ELSE, SWITCH, BREAK, CONTINUE, GOTO STATEMENTS, nested loops etc.) and writing programs using control structures; solving elementary programming problems from various areas of applications including mathematics and statistics.

UNIT-II:
Functions and program structure – Defining and accessing functions in C, passing arguments to a function, specifying argument data types – Illustration with example programs and problem solving through programs; Function prototypes, Functions returning non integers; Storage classes – Automatic, External, Static and Register variables, Scope rules, Header files, Block structure; Recursion in C – writing recursive programs and problem solving, The C Preprocessor

UNIT-III:
Definition and array processing, passing arrays to a function, multidimensional arrays, Arrays and Strings; POINTERS – pointers and addresses, pointer declaration, pointers and

function arguments – passing pointers to a function, Pointers and one dimensional arrays; Address arithmetic – operations on pointers, character pointers and functions; Pointer arrays/arrays of pointers, pointers to pointers, initialization of pointer arrays, pointers and multidimensional arrays; Command line arguments, Pointers to functions, passing functions to other functions.

UNIT-IV:
Structures and Unions – Basics of structures, processing of structures, user defined data types (typedef), Structures and Pointers, Structures and functions – passing structures to a function, Arrays of structures, Pointers to structures, Self-referential structures, Table lookup, UNIONS. writing programs and problem solving with structure and union

UNIT-V:
Input and output – Standard input and output, Formatted output – printf, Variable length argument, Formatted input - scanf; Data files – opening and closing data file, creating a data file, processing a data file, file access, unformatted data files, miscellaneous function in C; Low Level programming – Register variables, Bitwise operations, Bit fields, Enumeration, Commands Line arguments/parameters, Library functions, Macros, The C preprocessor.

Text Books:
1. The C Programming Language – Brian W. Kernighan and Denis M. Ritchie (PHI), Latest Edition
2. Theory and Problems of Programming with C – Byron S. Gottfried, (McGraw Hill), Latest Edition.
3. Programming with C - E. Balaguruswamy, McGraw Hill (Latest Edition)
4. Programming with C - Rajaraman R., PHI (Latest Edition)

Reference Books:
1. Let Us C – Yashavant P. Kanetkar, BPB Publications (Latest Edition)
2. Pointers in C – Yashavant P. Kanetkar, BPB Publications (Latest Edition)
3. Programming with ANSI C - B.T. Holmes, BPB (Latest Edition)

Course Code: CS-DSC-102 **Course Title: Digital Logic & Switching theory**

Credits: 3 **LTP: 3 – 0 – 0**

Course Objectives:

1. To explain the basics of digital logic and the concept of current flow in circuits.
2. To explain the different number systems.
3. To explain the binary, octal and hexadecimal operations.
4. To explain the design of the digital circuits using different gates.
5. To explain the combinational and sequential circuits.

Course Outcomes:

Students who complete the course will be able to do the following:

1. Convert any number from one number system to another.
2. Design any digital circuit.
3. Perform basic operations of gates.
4. Conceive the idea of memory in circuits.
5. Extend the digital logic to PLA, ROM etc.

UNIT-I:

Number Systems & Codes: Philosophy of number systems – complement representation of negative numbers, binary arithmetic- addition, subtraction, multiplication ,division, binary codes conversion-binary to decimal, binary to hexadecimal. Binary to octal, octal to binary, hexadecimal to binary, floating point representation.

UNIT-II:

Boolean Algebra and Switching Functions: Fundamental postulates of Boolean Algebra - Basic theorems and properties - switching functions–Canonical and Standard forms-Algebraic simplification digital logic gates, properties of XOR gates –universal gates-Multilevel NAND/NOR realizations.

Minimization of Switching Functions: Map method, Prime implicants, Don't care combinations, Minimal SOP and POS forms, Tabular Method, Prime –Implicant chart, simplification rules.

UNIT-III:

Combinational Logic Design: Design using conventional logic gates, Encoder, Decoder, Adders, Substractors Multiplexer, Demultiplexer, Modular design using IC chips, MUX Realization of switching functions Parity bit generator, Code-converters, Hazards.

UNIT-IV:

Sequential Circuits -I: Classification of sequential circuits (Synchronous, Asynchronous, Pulse mode, Level mode with examples) Basic flip-flops-Triggering and excitation tables. Steps in synchronous sequential circuit design. Design of modulo-N Ring & Shift counters, Serial binary adder, sequence detector.

UNIT-V:

Programmable Logic Devices, Threshold Logic: Basic PLD's-ROM, PROM, PLA, PLD Realization of Switching functions using PLD's. Capabilities and limitations of Threshold gate, Synthesis of Threshold functions, Multigate Synthesis.

TEXT BOOKS

1. Digital Logic and Computer Design- M. Morris Mano – Pearson Education, 2008.
2. Digital Design – Morris Mano, PHI, 3rd Edition, 2006.

REFERENCE BOOKS

1. An Engineering Approach to Digital Design – Fletcher, PHI, 2010.
2. Malvino A.P, Digital Principles and Applications, Tata McGraw Hill, 2009.

E – BOOKS

1. https://ebooks.lpude.in/computer_application/ad/DCAP108_DIGITAL_CIRCUITS_AND_LOGIC_DESIGNS.pdf
2. https://www.academia.edu/37310765/Digital_Logic_Design_4th_edition

MOOCs

1. https://nptel.ac.in/courses/117105080

2. https://www.udemy.com/course/digital-electronics-logic-design

Course Code: CS-DSM-101 **Course Title: Mathematics – I**
Credits: 3 **LTP: 3 – 0 – 0**

Course Objectives:
This course is designed to provide a deeper and rigorous understanding of fundamental concepts viz Set theory and functions, Matrix theory, Vector analysis, Differential Equations, Laplace transforms, Fourier Transforms and Integral Transforms etc. The main focus of this course will be on the theoretical foundation of the above said concepts and it will cultivate the rigorous Mathematical skills in the students.

Course Outcomes:
At the end of the course, the students will be able to:
1. Apply the knowledge of concepts of set theory and functions in order to further study and explore the concepts of Differential equations and other Mathematical concepts.
2. Apply Ordinary Differential Equations of various types, their solutions and fundamental concepts about their existence.
3. Apply various concepts of Matrix theory, Eigenvalues, Eigen vectors and their applications in Linear Algebra and Vector algebra.
4. Apply the concepts of Laplace Transforms and Fourier Transforms.

UNIT-I:
Set theory and Functions: set, subsets, union of sets, intersection of sets, difference of two sets, symmetric difference of two sets, van diagram, De-Morgan laws, distributive property,

Cartesian product, function, one-one function, onto function, bijective function, composition of functions and inverse function.

UNIT-II:
Matrix Theory: Symmetric and skew symmetric matrices, Hermitian and skew-Hermitian matrices, minor and cofactors, orthogonal and singular matrix, adjoint and inverse of matrices, application of matrices for solving linear system of equations, rank of a matrix, Eigenvalues, Eigen vectors, characteristic equation of a matrix, Cayley-Hamilton theorem and its use for finding inverse of matrices.

UNIT-III:
Vector Analysis: Introduction of vectors, Vector equation of straight lines, plane, circle and spheres, Scalar product of two vectors, Vector product of two vectors, scalar triple product, vector triple product, directional derivative, gradient of a scalar (function), divergence and curl of vectors, physical interpretation of gradient, divergence & curl, linear dependency and independency of vectors.

UNIT-IV:
Differential Equations: order and degree of differential equation, linear equations, Solution of first order linear differential equations by variable separable method, Homogeneous equations, linear equations, exact equations, Higher order linear differential equations with constant coefficients, Method of undetermined coefficients and Variation of parameters.

UNIT-V:
Integral transforms: Laplace transforms, Fourier series and Fourier transforms, related problems.

Text Books/ References:

1. Higher Algebra: Abstract & Linear- S.K. Mapa, Sarat Book House, 2003. (Unit I & II).
2. Vector Algebra: Shanti Narayan & PK Mittal, S. Chand & Co. Ltd.,2005. (Unit III).
3. Ordinary Differential Equations With Applications and Historical Notes: G.F. Simmons, Tata McGraw Hill, Second Edition. (Unit IV & V).

Course Code: CS-IDC-101 Course Title: Fundamentals of Computer & applications

Credits: 3 LTP: 3 – 0 – 0

Course objectives:
1. To introduce the concepts of Computer System to the students of UG level having no prior knowledge of Computer
2. To explain the concepts of basic units of Computer and its organization..

3. To explain data representation techniques and computer Software systems.
4. To explain working with windows operating system and MS Office

***Course Outcome*:**
After successful completion of the course, the students will be able to:
1. Apply the concepts of Computer System, its functions and working
2. Apply the basic units of Computer system and its organization
3. Apply about operating systems and its working
4. Work with windows Operating System and various software packages of MS Office

UNIT 1:
Introduction to Computer System: Introduction to Computer System: Definition of Computer, and Basic Characteristics and Operations performed by Computers, History and Generations of Computers, Classification of Computers.

Basic Organization of Computer Systems: Fundamental model (Von-Neuman) and Block diagram of Computer: ALU, CU, CPU; Input, Output and Storage Units and their Functions; Types of storages and storage devices used in Computer Systems, Input and Output Devices.

Introduction to Number Systems – Decimal, Binary, Octal and Hexadecimal number systems and conversion from one system to another

UNIT II:
Data Representation and Computer Software Systems: Data Representation**:** Representation of data in Binary and data types: Numeric, Alphabetic and Alphanumeric; Computer codes: Most commonly used computer codes – Bits and Bytes; BCD, EBCDIC, ASCII and UNICODE; Introductory concept of Boolean Algebra and Basic Gates: AND, OR, NOT, NAND, XOR.

Computer Software Systems – A mechanism for Human Computer Interface; Concept of Software and Types of Software: System Software, Application Software, Firmware, Middleware. Software Development Steps. Computer Languages and its Types: Machine Language, Assembly Language, High Level Languages: advantages and disadvantages of computer languages. Concept of Translators: Compiler, Linker, Loader, Interpreter and Operating System.Concept of Computer Programs Design and Development: Algorithms, Flowchart, Pseudocodes, Control Structures, Illustrative Examples

UNIT III:
Operating Systems, Introduction to Operating Systems: Types of Operating Systems: Uniprocessor and Multiprocessor Single user and Multiuser, Overview of Functions of Operating System – Process management, Memory management, File management, Processor and Device Management, Security and Command Interpretation, Overview of UNIX and LINUX Systems. Introduction to computer networking

Windows Operating System: Introduction to Windows, Starting Windows, Desk Top, Task Bar, Start Up Menu Working with programs and icons-Adding, removing, starting and quitting programs and icons. Working with files and folders-creating, deleting, opening, finding, copying, moving and renaming files and folders. Control Panel, setting, My Computer, Recycle bin, My documents, drives. Windows notepad, Accessories and Windows Explorer.

UNIT IV:

Word Processor and SpreadSheet, MS-Word: Overview of Word Processing, Word Processor and its features, Parts of word window, Types of Menus, Opening, creating saving, cut, copy and paste, save & save as. Editing of Text, Find and Replace, print and print preview. Word count, Bullets and Numbering, Spell Checker, Grammar Checker, Auto Correct, Auto Complete, Auto Text, Header and footer, tables, mail merge, border and shading, page setup, printing. Mail Merge, Table handling and important shortcut keys, Macros.

Spreadsheet: MS Excel and its features, Entering Information in Worksheet, Editing Cell Entry, Moving and Copying Data, deleting or Inserting Cells, Rows and Columns, Custom Numeric Formats, Using Formulas and functions, Creating charts, Sorting and Searching.

UNIT V:

Presentation Software and Usage , MS-PowerPoint: Presentation Softwares and its uses, Overview of MS-PowerPoint, Steps for creating PowerPoint Presentation, Slides, PowerPoint Views, Auto content, Assigning Slide Transitions, Using Preset Animations, Hiding Slides, Slide Show, Controlling the Slide Show with a Keyboard, Setting Slide Show Timings, Wizard, Custom Animation, Transition and build effects, Printing slides and important shortcut.

Use of Computers in Education and Research: Data analysis, e-Library, Google Scholar

Text Books:

1. P.K Sinha & Priti Sinha, Computer Fundamentals, BPB Publications, (Latest Edition)
2. V. Rajaraman, Fundamentals of Computers, 6 th edition PHI Learning Private Limited 2014 (Latest Edition)
3. R.K. Taxali, PC Software for Windows (Latest Edition)

Reference Books:

1. Alexis Leon and Matthews Leon: Introduction to Computers, Leon Vikas,1999.
2. Suresh K. Basandra, Computer Systems Today, Galgotia Publications.
3. Peter Norton: Computing Fundamentals. 6 th Edition, McGraw Hill-Osborne,2007
4. Joyce Coax , Joan Preppernau,,Steve Lambert and Curtis Frye,2007
5. Microsoft Office System step by step, Microsoft Press (Latest Edition)
6. Microsoft Office 20xx Training Guide, BPB Publications (Latest Edition)

Course Code: CS-AEC-I MIL-101 **Course Title: MIL-I**

Credits: 2 **LTP: 2 – 0 – 0**

Course Code: CS-SEC-101 **Course Title: a) Lab on programming in C**

Credits: 3 **LTP: 0 – 0 – 6**

Course Objectives:

1. To explain design and implementation of C programs
2. To explain writing and executing programs with control structures and functions
3. To explain writing and executing programs with pointers in C
4. To explain writing and executing programs with structures and unions
5. To explain writing and executing programs with files in C

Course outcomes:

After successful completion of the course students will be able to:

1. Write programs and implement programs in C.
2.Write programs and execute programs with control structures and functions
3. Write and execute programs with pointers in C
3. Write and execute programs with structures and unions
4. Write and execute programs with files in C

Problem solving of various nature by implementing programs in C Programming languages based on unit wise contents of the theory paper Programming with C. Following are some programming tasks for laboratory programming assignments but the assignments are not limited to these only.

List of laboratory programming assignments (not limited to these):

1. Write a program to

   a) Produce ASCII equivalent of given number

   b) Find the divisor or factorial of a given number.

   c) Evaluate the following algebraic expressions after reading necessary values from the user (ax+b)/ (ax-b) -2.5 log x-cos 30+|x2-y2|+sqrt (2xy) - (x5+10x4+8 x 3+4x+2)

   d) Find sum of a geometric series

e) Cipher a string

f) Check whether a given string follows English capitalization rules

g) Find sum of the following series 1+ ½ + 1/3 +_____+1/20

h) Search whether a given substring exist in an input string or not and then delete this string from input string.

2. Write a recursive program for tower of Hanoi problem

3. The Fibonacci sequence of numbers is 1, 1, 2, 3, 5, 8……. Based on the recurrence relation F(n)=F(n-1)+F(n-2) for n>2 Write a recursive program to print the first n Fibonacci number

4. Write a menu driven program for matrices to do the following operation depending on whether the operation requires one or two matrices

a) Addition of two matrices

b) Subtraction of two matrices

c) Finding upper and lower triangular matrices

d) Trace of a matrix

e) Transpose of a matrix

f) Check of matrix symmetry

g) Product of two matrices.

5. Write a program that takes two operands and one operator from the user perform the operation and then print the answer

7. Write functions to add, subtract, multiply and divide two complex numbers (x+iy) and (a+ib) Also write the main program.

8. Write a menu driven program for searching and sorting with following options:-

a) Searching (1) Linear searching (2) Binary searching

b) Sorting (1) Insertion sort (2) Selection sort

9. Write a program to copy one file to another, use command line arguments.

10. Write a program to mask some bit of a number (using bit operations)

11. An array of records contains information of managers and workers of a company. Print all the data of managers and workers in separate files.

Course Code: CS-SEC-101 Course Title: b) Lab on Digital Logic & switching theory

Course Objectives:

1. To demonstrate the basics of digital logic and concept of current flow in circuits.
2. To demonstrate the different number (binary, oct, hex) conversion.
3. To demonstrate the basic truth tables and gate operations.
4. To demonstrate the combinational circuits.
5. To demonstrate the sequential circuits.

Course Outcomes:

After successful completion of the course, the students will be able to:

1. Convert any number from one number system to another.
2. Design any digital circuit.
3. Perform basic operations of gates.
4. Conceive the idea of memory in circuits.
5. Extend the digital logic to PLA, ROM etc.

Following are some samples for laboratory assignments but the assignments are not limited to these only:

01: To study the digital board DB-01 and to verify the truth tables for AND, OR, NOT, NAND, NOR and XOR gates.

02: To study the digital board DB-02 and to verify the truth tables for Universal gates (NAND, NOR).

03: To study the digital board DB-03 and to implement EX-OR gate.

04: To study the digital board DB-04 and analyze De-Morgan's Theorem.

05: To study the digital board DB-05 and to study the Ex-OR gate implementation for

   a. Odd parity generator
   b. Even parity generator
   c. Binary word comparator

06: To study the digital board DB-06 and to study the code conversion circuits for

   a. Binary to gray code
   b. Gray to binary code

07: To study the digital board DB-07 and to verify BCD to Excess 3 code conversion circuit.

08: To study the digital board DB-09 and to verify the truth tables for

   a. 8 to 3 line encoder
   b. 3 to 8 line decoder

09: To study the digital board DB-10 and to verify the truth tables for Multiplexer(MUX) &DeMultiplexer (DMUX).

10: To study the digital board DB-11 and to verify the truth tables for the R-S flip flops, D flip flops, J-K flip flops & T flip flops.

TEXT BOOKS

1. Digital Logic and Computer Organisation- V. Rajaraman & T. Radhakrishnan–PHI, 2006.
2. Digital Logic Fundamentals – Ananthi S. and J.G. Sheshasaayee, Margham Publications, 2010.

REFERENCE BOOKS

1. An Engineering Approach to Digital Design – Fletcher, PHI, 2010.
2. Malvino A.P, Digital Principles and Applications, Tata McGraw Hill, 2009.

E – BOOKS

1. https://ebooks.lpude.in/computer_application/ad/DCAP108_DIGITAL_CIRCUITS_AND_LOGIC_DESIGNS.pdf
2. https://www.academia.edu/37310765/Digital_Logic_Design_4th_edition

MOOCs

1. https://nptel.ac.in/courses/117105080
2. https://www.udemy.com/course/digital-electronics-logic-design

Course Code: CS-VAC-101 **Course Title: NCC/NSS/Sports/HW/UI/GCS/Yoga**

Credits: 3

Semester II

Course Code: CS-DSC-151 **Course Title: Python Programming**

CREDIT: 3 **L-T-P: 3-0-0**

Course Objectives:
To understand why Python is a useful scripting language for developers; to learn how to use lists, tuples, and dictionaries, sets in Python programs; to learn how to use indexing and slicing to access data in Python programs ; to learn how to write loops and decision statements in Python; to learn how to write functions and pass arguments in Python; to learn how to build and package Python modules for reusability; to learn how to read and write files in Python; to learn how to design object-oriented programs with Python classes ; to learn how to use class inheritance in Python for reusability; to learn how to use exception handling in Python applications for error handling ; to learn how to use Machine Learning tools like Pytorch, keras, TensorFlow etc.

Course Outcomes:
At the end of the course it is expected that a student would be reasonably proficient in writing Python programs for solving various problems as the course covers topics ranging from basics of Python Programming to advanced level.

UNIT-I
Introduction: Basic Elements of Python, Operators, Python Statements & Comments, Python Type Conversion, Indentation in Python, print() and input() functions, Strings, Python IDEs
Python Flow Control: if…else , for loop, while loop, break and continue, Python Pass, range statement; Python memory model: names, mutable and immutable values

UNIT-II
Python Functions: Python Functions, Function Arguments, Recursion, Inductive function definition, Anonymous Function, Lambda function, Passing functions as arguments, Python Global Keyword, Python Modules, Python Package.

UNIT-III
Python Collections: List, Tuple, Sets and Dictionary; String Manipulation: Basic Operations, Slicing, Python Regular expressions; Python iterators, Python Generators, Python Closure, Python Decorators, Higher order functions on lists: map, filter, list comprehension

Unit IV
Python Files: Python File Operation, Python Directory, Python Exception Handling, User defined exception; Assertions; Classes and objects in Python; MATPLOTLIB

Unit V

Arrays vs lists, Scope in Python: local, global, nonlocal names, Nested functions, Binary Search, Data structures: stack, queue implementation in Python, Sorting: Merge sort , Quicksort, Stable sorting implementation in Python, Linked lists: find, insert, delete, Binary search trees: find, insert, delete

Text Books:

1. Introduction to Computation and Programming Using Python, John V. Guttag, PHI
2. Core Python Programming, Dr. R.Nageswara Rao, Dreamtech Press

Reference:

1. Swayam course on "Programming, Data Structures And Algorithms Using Python" By Prof. Madhavan Mukund, Chennai Mathematical Institute

Course Code: CS-DSC-152 **Course Title: Numerical Methods**

CREDIT: 3 **L-T-P: 3-0-0**

Course Objectives:

The objective of this course is to familiarize the various numerical methods to solve scientific problems. This course covers the designing and understanding of iterative algorithms to solve numerical problems.

Course Outcomes:

At the end of this course student will be able to:

1. Apply the different types of errors in numerical methods.
2. Apply the designing of iterative algorithms to solve various numerical problems.
3. Able to apply the basic numerical methods to to solve nonlinear equations, set of linear equations, numerical approximation methods and numerical integration and numerical differentiation.
4. Able to apply the uses and importance of iterative algorithms to solve numerical problems

UNIT-I:

Approximation and Error in Computing**:** Introduction, Significant digits, different types of error, Absolute and relative error, Error estimation, Floating point arithmetic and Round off error. Solution of nonlinear equations: Bisection's method, Newton-Raphson's Method, Secant's Method.

UNIT-II:

Interpolation: Introduction, Errors in Polynomial Interpolation, Finite differences: Forward

Differences- Backward differences, Differences of a polynomial, Newton's formula for interpolation, Central difference interpolation Formula – Gauss Central Difference Formula, Lagrange's Interpolation formula.

UNIT-III:
Solution of linear system: Matrix inversion method, Gauss Elimination, Gauss-jordan method. Curve Fitting: Fitting a straight line, Second degree curve-exponential, curve-power curve by method of least squares.

UNIT-IV:
Numerical Differentiation and Integration: Cubic spline method, Trapezoidal rule – Simpson's 1/3 Rule –Simpson's 3/8 Rule. Partial differential equation: Laplace's equation-Jacobi's Method- Gauss seidel method.

UNIT-V:
Numerical solution of Ordinary Differential equations: Solution by Taylor's series-Picard's Method of successive Approximations-Euler's Method-Runge-Kutta Methods, Predictor-Corrector Methods, Adams- Moulton Method, Milne's Method.

Text Books/References:

1. Balagurusamy, E., "Numerical Methods", Tata McGraw-Hill Pub.Co.Ltd, New Delhi, 1999.
2. D.Kincaid, W.cheney,"numerical Analysis",Brooks/Cole Publishing Company, California,2001.
3. S.S. Sastry, "Introductory Methods of Numerical Analysis", PHI learning Pvt Ltd, 5thEdition 2010.
4. Computer Oriented Numerical Methods - Rajaraman, PHI
5. Numerical Computations - Venkataraman

Course Code: CS-DSM-151 **Course Title: Mathematics-II**

Credit: 3 **LTP: 3-0-0**

Course Objectives :
This course is designed to provide a deeper and rigorous understanding of Real Analysis. The main focus of the course will be on the theoretical foundation of Differential and Integral Calculus.

Course Outcomes:
At the end of the course, the students will be able to :
1. Apply the nature of abstract Mathematics and explore the concepts in further details .
2. Identify challenging problems in real variable theory and find their appropriate

solutions .

3. Deal with axiomatic structure of Differential Calculus and generalize the concepts of sequences, series, and continuous functions.

UNIT-I:

Differential Calculus: Limit, Cauchy's criteria for existence of limits (without proof), problems on limit, Continuity: $\varepsilon-\delta$ definition of continuity, problems on continuity, bounded functions, properties of continuous and bounded functions.

UNIT-II:

Differentiability: $\varepsilon-\delta$ definition of derivative, relation between continuity and differentiability, Intermediate value theorem, Rolle's Theorem, mean value theorem, L'Hospital rule and Taylor's theorem.

UNIT-III:

Successive differentiation, Leibnitz theorem, curvature, asymptotes, singular points, Functions of several variables: partial derivative, total differentials, Euler's theorem of homogeneous function of two variables, Jacobian, maxima, minima, necessary and sufficient conditions for maxima and minima.

UNIT-IV:

Sequence: Convergent sequence, monotone sequence, subsequences and Bolzano-Weierstrass theorem, Cauchy criterion for convergence of sequence and divergent sequences.

Series: Convergence and absolute convergence of series, limit comparison test, root test, ratio test, integral test, raabe's test, alternating series test and tests for non absolute convergence of a series.

UNIT-V:

Integral Calculus: definition and properties of definite integrals, Riemann integrable functions, Fundamental theorem, Area bounded by plane curves, Volumes and surfaces of solid of revolution about axis.

Text Books:

1. Differential Calculus: Das & Mukherjee, U.N. Dhur Publishers, 1975. (Unit -I, II, III, IV)
2. Integral Calculus: Das & Mukherjee, U.N. Dhur Publishers, 1998. (Unit – V)

Reference:

1. Introduction To Real Analysis: Bartle & Sherbert, Wiley Student Edition, Third edition.

Course Code: CS-IDC-151 Course Title: Introduction to Internet Technology

CREDIT: 3 **L-T-P: 3-0-0**

Course Objectives:

1. Introduction of Internet and basic components of Internet.
2. To explain the fundamentals of HTML and HTML 5.
3. To explain the basic properties of CSS and CSS3 in html file
4. To explain the basic properties of JavaScript in html file
5. To explain dynamic web page using JavaScript

Course Outcomes:

Upon successful completion of the course, the students will:

1. Apply the basics of Web Designing & Publishing.
2. Apply the implementation of HTML tags and be able to create HTML static web pages.
3. Able to use basic properties of JavaScript in html file
4. Able to create dynamic web page using JavaScript

UNIT I:

BASICS OF WEB DESIGN: Overview of Internet and WWW, Basic elements of the Internet, Internet services, Internet Browsers and Servers, Introduction to WWW, URL, webpage, web site, web
servers, web browser, Web Application, Client and Server-side scripting languages, types of websites, Web Design and Development, Internet Addressing: standard Internet Address, Domain Name Server (DNS).

UNIT II:

WEB PAGE DEVELOPMENT USING HTML5: HTML Fundamentals: HTML & its relevant history, Anatomy of an HTML Tag, Basic HTML Document Structure, working with Text, working with Lists, Tables and Frames,working with Hyperlinks, Images and Multimedia, Working with Forms and Controls. Advanced Elements in HTML5: Semantic Elements, New Input Type Elements, Multimedia Tags.

UNIT III:

INTRODUCTION TO CASCADING STYLE SHEETS: Introduction to Cascading Style Sheet (CSS), basic syntax and structure, CSS selectors, Ways of specifying style, CSS Properties, CSS Styling (Background, Text, Fonts, Lists, Tables, Links), CSS Box Model, CSS Navigation Bar; CSS3: - CSS Rounded Corners, Box & Text Shadow, Gradients, Background Images, Transitions, Transforms, and Animations, CSS Layout.

UNIT IV:

JAVA SCRIPT PROGRAMMING I: Introduction to Client-Side Scripting, Basics of Java Script, Java Script Statements, Comments, variables, Operators and Expressions, Conditions

statement, Functions, Dialog boxes.

UNIT V:
JAVASCRIPT PROGRAMMING II**:** The Java Document Object Model (DOM): JavaScript Document Object Model hierarchy – Create, find and manipulate HTML Element using Objects and methods. Form validation, Applying Style using JavaScript. Creating New window, Accessing & manipulating, History of HTML Pages; Forms: Form object, built in objects, User defined objects, Cookies, Java Script Window; DHTML: Introduction to DHTML, DHTML CSS, DHTML Java Script, DHTML HTML DOM, DHTML Events.

Text Books:
1. Matthew MacDonald: HTML5 The Missing Manual, O'Reilly Media, August 2011.
2. Peter Gasston: The Book of CSS3, A Developer's Guide to the Future of Web Design, No Starch Press, April 2011.
3. Richard York: Beginning CSS Cascading Style sheets for Web Design, Wrox Press (Wiley Publishing), 2005.

Reference Books:
1. **Ivan** Bayros: Web Enabled Commercial Application Development using HTML, JavaScript, DHTML and PHP, fourth revised edition, BPB Publication.
2. David Mc Farland: CSS The Missing Manual, O'Reilly, 2006.
3. Julie C. Meloni: HTML, CSS and JavaScript All in One, Pearson.

Web References:
1. http://www.tutorialspoint.com/html5 [For notes on HTML5 tags]
2. http://www.w3schools.com/html/ [For HTML5, CSS and JavaScript notes and examples]
3. https://in.godaddy.com/help/dreamweaver-cs6-publish-your-website-7811[Publish your website using Dreamweaver]
4. http://fullbooksfreedownload.blogspot.in/2016/02/html-css-javascript-webpublishing-in.html [Book :- HTML, CSS & JavaScript Web Publishing in One Hour a Day, Sams Teach Yourself, 7th Edition PDF]

Course Code: CS-AEC-I EL-151 **Course Title: ENGLISH-1**

Credits: 2 **LTP: 2– 0 – 0**

Course Code: CS-SEC-151 **Course Title: b) Lab on Python Programming**

Credits: 3 **LTP: 0– 0 – 6**

Course Objectives: The objective of the course is to make the students aware and proficient in Python Programming**.**

Course Outcome: The outcome of the course is that the students should be able to execute the broad class of problems as outlined in the syllabus of the lab on python programming.

1. Problems related to if ... .else structure of Python
2. Problems related to looping, break and continue
3. Problems to identify the usage of pass and range statements in python.
4. Problems related to usage of functions in Python, Global , local and Non local functions
5. Problems related to Recursion, Anonymous Function, Lambda function, Python Modules and Python Package.
6. Problems related to File handling in Python, exception handling, usage of User defined exception and Assertions
7. Problems related to Python Collections: List, Tuple, Sets and Dictionary
8. Problems related to String Manipulation, Basic Operations, Slicing, Python Regular expressions; Python iterators, Python Generators, Python Closure, Python Decorators, List Comprehension
9. Problems related to Python Object & Class , Problems on MATPLOTLIB
   Stack, queue implementation in Python, Implementing basic sorting algorithms like Merge Sort, Quick sort and stable sort
10. Problems related to Binary Search, Linked List and Binary Search trees

Course Code: CS-SEC-151 Course Title: b) Laboratory on Numerical Methods

Course Objectives:

The objectives of this course are to design the iterative algorithms and execute in the computer to solve various scientific numerical problems.

Course Outcomes:

Upon successful completion of the course, the students will:

1. Apply the structure of iterative algorithms that can be converted into computer program code.
2. Able to execute various numerical methods in the computer and will get a better understanding of these methods.
3. Improvement in the programming skills of the students

Problems related to scientific computation should be solved by using the high level programming language C (preferably on Unix/Linux/Solaris operating systems on a network). Following are some sample laboratory programming assignments but the assignments should not be limited to these only:

Write an algorithm and a C program to:

1. Find the roots of a given equation using:
   a. Bisection Method.
   b. Regula Falsi Method.
   c. Newton Raphson Method.
   d. Secant Method.
2. Find f(x) for a given set of experimental data using
   a. lagrange interpolation.
   b. Newton's forward interpolation.
   c. Newton's backward interpolation.
3. Fit a given set of data in a
   a. straight line
   b. parabola
   c. curve of the form $y=ax^2+b$
   d. curve of the form $y=ab^x$
   e. curve of the form $y=ae^{bx}$
4. Find the numerical solution of a system of linear equation using
   a. Gauss elimination
   b. Gauss Jacobi
   c. Matrix Inversion
   d. Gauss Seidel
5. Perform differentiation applying
   a. Taylor Series
   b. Euler's method
   c. Runge-Kutta method
   d. applying Laplace's equation
   e. Jacobi's Method
   f. Gauss seidel method
   g. Trapezoidal Rule
6. Perform numerical Integration applying Simpson's 1/3 Rule.

Course Code: CS-VAC-151-NCC/NSS/EVS **Course Title: EVS**

Credits: 3 **LTP: - - -**