

# **Assam University, Silchar**



## **Four Year Undergraduate Programme**

**Implemented under NEP 2020**

**Effective from the Academic Year 2023-24**

## **Syllabus of Computer Science (4<sup>th</sup> Year)**

**Approved in the 99th meeting of the Academic Council on 22.05.2025 vide Resolution No AC:99:05-25:5**

**Semester wise list of Computer Science Discipline Specific Core (DSC) Papers**

<b>SEMESTER</b>	<b>COURSE CODE</b>	<b>TITLE OF COURSES</b>	<b>CREDITS</b>
<b>I</b>	<b>CSCDSC101</b>	Digital Computer Fundamentals	3
	<b>CSCDSC102</b>	Discrete Mathematics	3
<b>II</b>	<b>CSCDSC151</b>	Data Structure	3
	<b>CSCDSC152</b>	Lab on Data Structure	3
<b>III</b>	<b>CSCDSC201</b>	Computer Organization and Architecture	4
	<b>CSCDSC202</b>	Operating Systems	4
<b>IV</b>	<b>CSCDSC251</b>	Object Oriented Programming with Java	4
	<b>CSCDSC252</b>	Database Management System	4
	<b>CSCDSC253</b>	Lab on Java & DBMS	4
<b>V</b>	<b>CSCDSC301</b>	Computer Graphics	4
	<b>CSCDSC302</b>	System Analysis and Design & Cyber Security	4
	<b>CSCDSC303</b>	Lab on Graphics Programming	4
<b>VI</b>	<b>CSCDSC351</b>	Computer Network and Internet Technology	4
	<b>CSCDSC352</b>	Theory of Computation	4
	<b>CSCDSC353</b>	Microprocessor and Systems Programming	4
	<b>CSCDSC354</b>	Lab on Internet Technology & Microprocessor and Systems Programming	4
<b>VII</b>	<b>CSCDSC401</b>	Design and Analysis of Computer Algorithms	4
	<b>CSCDSC402</b>	Principles of Compiler Design	4
	<b>CSCDSC403</b>	Artificial Intelligence	4
	<b>CSCDSC404</b>	Lab on DACA & Compiler Design	4
<b>VIII</b>	<b>CSCDSC451</b>	Software Engineering	4
	<b>CSCDSC452(A)</b>	Image Processing	4
	<b>CSCDSC452(B)</b>	Data Analytics	4
	<b>CSCDSC453</b>	Natural Language Processing	4
	<b>CSCDSC454(A)</b>	Internet of Things	4
	<b>CSCDSC-454(B)</b>	Cloud Computing	4
	<b>OR</b>		
	<b>CSCDSC451</b>	Research Methodology	4
	<b>CSCDSC455</b>	Research Project/Dissertation	12

**Semester wise list of Computer Science Discipline Specific Minor (DSM) Papers**

<b>SEMESTER</b>	<b>COURSE CODE</b>	<b>TITLE OF COURSES</b>	<b>CREDITS</b>	<b>DSM1/DSM2</b>
<b>I</b>	<b>CSCDSM101</b>	Programming with C	3	<b>DSM1</b>
<b>II</b>	<b>CSCDSM151</b>	Programming with C	3	<b>DSM2</b>
<b>III</b>	<b>CSCDSM201</b>	Database Management System	4	<b>DSM1</b>
<b>IV</b>	<b>CSCDSM251</b>	Lab on C & DBMS	3	<b>DSM1</b>
	<b>CSCDSM252</b>	Database Management System	3	<b>DSM2</b>
<b>V</b>	<b>CSCDSM301</b>	Operating Systems	3	<b>DSM1</b>
	<b>CSCDSM302</b>	Operating Systems	3	<b>DSM2</b>
<b>VI</b>	<b>CSCDSM351</b>	Lab on C & DBMS	4	<b>DSM2</b>
<b>VII</b>	<b>CSCDSM401</b>	Internet Technologies	4	<b>DSM1</b>
<b>VIII</b>	<b>CSCDSM451</b>	Internet Technologies	4	<b>DSM2</b>

**Semester wise list of Computer Science Skill Enhancement Course (SEC) Papers**

<b>SEMESTER</b>	<b>COURSE CODE</b>	<b>TITLE OF COURSES</b>	<b>CREDITS</b>
<b>I</b>	<b>CSCSEC101</b>	Programming with C	3
<b>II</b>	<b>CSCSEC151</b>	Python Programming	3
<b>III</b>	<b>CSCSEC201</b>	Programming with C++ & Lab on OS and C++	3

**Semester wise list of Computer Science Interdisciplinary Course (IDC) Papers**

<b>SEMESTER</b>	<b>COURSE CODE</b>	<b>TITLE OF COURSES</b>	<b>CREDITS</b>
<b>I</b>	<b>CSCIDC101</b>	Computer Fundamentals & Applications	3
<b>II</b>	<b>CSCIDC151</b>	Programming Fundamentals with C	3
<b>III</b>	<b>CSCIDC201</b>	Introduction to Web Designing & Cyber Security	3

<b>Semester</b>	<b>: VII</b>
<b>Course Type</b>	<b>: DSC</b>
<b>Course Code</b>	<b>: CSCDSC401</b>
<b>Name of the Course</b>	<b>: Design and Analysis of Computer Algorithms</b>
<b>Learning level</b>	<b>: Advanced Course</b>
<b>Credits</b>	<b>: 4</b>
<b>Contact Hours</b>	<b>: 60</b>
<b>Total Marks</b>	<b>: 100</b>
<b>End Semester Marks</b>	<b>: 70</b>
<b>Internal Marks</b>	<b>: 30</b>

*Course Objectives: The course objective is to*

- 1. Learn fundamental techniques for designing efficient algorithms, including divide and conquer, greedy methods, dynamic programming, and backtracking*
- 2. Analyzing Algorithm Efficiency using asymptotic notation (Big O, Omega, Theta)*
- 3. Study graph algorithms such as BFS, DFS, shortest path and minimum spanning tree (Kruskal, Prim).*
- 4. Understand P vs NP problems, reducibility, and heuristic approaches for solving complex problems.*

#### **UNIT-I**

**Introduction:** Writing structured programs, Analyzing Algorithms, Stacks and Queues, Trees, Heaps and Heap sort, Graphs, Hashing.

#### **UNIT-II**

**Divide and Conquer:** The general concept, Binary Search, Finding the maximum and minimum, Merge sort, Quick sort, Selection sort, Strassen's matrix manipulation, Greedy methods, Minimum spanning trees.

#### **UNIT-III**

**Dynamic Programming:** The General method, Multi stage graphs, All pairs shortest paths, Optimal binary search trees, Reliability design, The Travelling sales person Problem, Flow shop scheduling.

#### **UNIT-IV**

**Basic search and traversal techniques:** The techniques, Code generation, AND/OR graphs, Game trees, Biconnected components, Depth first search and Breadth first search

#### **UNIT-V**

**Backtracking:** the queens problems, Introduction to NP- Hard and NP- Complete problems.

*Course Learning Outcomes: After successful completion of the course, the students will be able to*

- 1. Analyze Algorithm Efficiency i.e. evaluate the time and space complexity of algorithms using asymptotic notations (Big O,  $\Omega$ ,  $\Theta$ ) and differentiate between worst-case, best-case, and average-case complexities.*
- 2. Apply Algorithmic Design Techniques*

3. *Develop algorithms using techniques such as divide and conquer, greedy algorithms, dynamic programming, backtracking, and branch & bound.*
4. *Identify the appropriate algorithmic approach for solving different types of problems.*
5. *Understand NP-Completeness and Computational Complexity i.e. distinguishes between P, NP, NP-complete, and NP-hard problems.*

#### **Text Books:**

1. S. Sahni, and Horowitz, Rajasekaran, Computer Algorithm, 7<sup>th</sup> Edition, Galgotia Publications, 2010
2. Horowitz and Sahni, Fundamentals of Computer Algorithm, 6<sup>th</sup> Edition, Galgotia Publications, 2012
3. Aho, Hopcroft and Ullman, Design and Analysis of Computer Algorithm, 3<sup>rd</sup> Edition, Addison-Wesley, 2014.
4. S. E Goodman and S. T Hedet, Introduction to the Design and Analysis of Algorithm, 4<sup>th</sup> Edition, 2009.

#### **Reference Books:**

1. S.K Basu, Design Methods and Analysis of Algorithms, 2<sup>nd</sup> Edition, PHI, 2005.
2. Gilles Brassard and Paul Brantley, Algorithmic theory and Practices, 3<sup>rd</sup> Edition, PHI, 2007.
3. Knuth, Fundamental Algorithms, The Art of Computer Programming, Vol-1, 3<sup>rd</sup> Edition, Addison-Wesley, 1997.
4. S. Sahni, Analysis of Algorithms through C++, 4<sup>th</sup> Edition, Tata McGraw Hill, 2009.

<b>Semester</b>	<b>: VII</b>
<b>Course Type</b>	<b>: DSC</b>
<b>Course Code</b>	<b>: CSCDSC402</b>
<b>Name of the Course</b>	<b>: Principles of Compiler Design</b>
<b>Learning level</b>	<b>: Advanced Course</b>
<b>Credits</b>	<b>: 4</b>
<b>Contact Hours</b>	<b>: 60</b>
<b>Total Marks</b>	<b>: 100</b>
<b>End Semester Marks</b>	<b>: 70</b>
<b>Internal Marks</b>	<b>: 30</b>

*Course Objectives: The course objective is to*

1. Understand to compiler basics, including lexical analysis and syntax trees.
2. Understand various phases in the design of the compiler
3. Introduce the Top-Down and Bottom-Up parsing algorithms.
4. Introduce symbol table and run-time storage management.
5. Explain code generation and code optimization.

#### **Unit-I**

**Overview of compiling process, some typical compiler structures, Symbol Table:** The contents of a symbol table, Data structures for symbol tables (ST), design of an ST, The Phases of a Compiler.

#### **Unit-II**

**Lexical Analysis:** Role of Lexical Analyzer, Input Buffering, Specification of Tokens, Recognition of Tokens, Regular Expressions.

### **Unit-III**

**Syntax Analysis:** Need and role of the Parser. Context Free Grammars, Derivation and Parse Tree, Parsing Techniques- Top-down parsing: Recursive - Descent parsing, LL(1) parser. Bottom-up parsing: Shift-Reduce Parser, Operator precedence parser, LR, parser, SLR parser, LALR parser, CLR parser.

### **Unit-IV**

**Intermediate Code Generation:** Intermediate Languages, Intermediate Representation Techniques, Statements in three-address code, Implementation of three-address instruction, Three-address code generation, Code Generation for arrays, Translation of Boolean expressions, Translation of Control flow statements, Translation of Case Statements, Function Calls.

### **Unit-V**

**Code Generation:** Basic block, Representation of Basic blocks, Problems in Code Generation, Code Generation for DAGs, Register Allocation and Register Interference Graph, Code Generation using Dynamic Programming. **Code Optimization:** Principal sources of optimization, loop optimization, Global data flow analysis.

**Course Learning Outcomes:** *After successful completion of the course, the students will be able to*

1. Understanding lexical analysis and syntax trees is essential for understanding compiler basics.
2. Understanding top-down and bottom-up parsing algorithms.
3. An understanding of the symbol table and how run-time storage is handled.
4. Getting a better understanding of intermediate code generation.
5. Understanding the generation and optimization of code.

### **Text Books:**

1. Alfred V Aho, R.Sethi, D. Ullman, Compilers - **Principles, Techniques and tools**, Pearson, 2<sup>nd</sup> Edition, 2014
2. Santanu Chattopadhyaya, “**Compiler Design**”, PHI, 2<sup>nd</sup> Edition, 2022.

### **Reference Books:**

1. Y. N. Srikant, & P. Shankar, “**The Compiler Design Handbook: Optimizations and Machine Code Generation**”, CRC Press, 2<sup>nd</sup> Edition, 2018.
2. K. Muneeswaran, “**Compiler Design**”, Oxford University Press, 2012

<b>Semester</b>	<b>: VII</b>
<b>Course Type</b>	<b>: DSC</b>
<b>Course Code</b>	<b>: CSCDSC403</b>
<b>Name of the Course</b>	<b>: Artificial Intelligence</b>
<b>Learning level</b>	<b>: Advanced Course</b>
<b>Credits</b>	<b>: 4</b>

<b>Contact Hours</b>	<b>: 60</b>
<b>Total Marks</b>	<b>: 100</b>
<b>End Semester Marks</b>	<b>: 70</b>
<b>Internal Marks</b>	<b>: 30</b>

*Course Objectives: The course objective is to*

1. Introduce the historical evolution of Artificial Intelligence (AI), its foundations and applications.
2. Explain different search techniques.
3. Introduce the concepts of propositional and predicate logic and their applications in AI.
4. Explore the scope, potential, limitations, and implications of intelligent systems

### **Unit-I**

**Introduction:** Introduction to Artificial Intelligence, History of AI, Risks and Benefits of AI, Brief discussion of applications of AI (Expert System, Natural Language Processing, Speech and Pattern Recognition etc.), Importance of AI. Introduction to Intelligent Agents, their structure, behavior and environment.

### **Unit-II**

#### **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.

### **Unit-III**

**Knowledge Representation:** Introduction to First Order Predicate Logic, Resolution Principle, Unification, Semantic Nets, Conceptual Dependencies, Frames, and Scripts, Production Rules, Conceptual Graphs. Various approaches used in knowledge representation, Issues in knowledge representation.

### **Unit-IV**

**Machine Learning:** Forms of learning - Supervised Learning, Unsupervised Learning, Reinforcement Learning; Learning Probabilistic Models, Deep Learning.

### **Unit-V**

**Application areas and AI languages:** Applications of AI in Engineering, AI languages and their important characteristics, problem solving using Programming in Logic (PROLOG).

*Course Learning Outcomes: After successful completion of the course, the students will be able to:*

1. Demonstrate fundamental understanding of the history of Artificial Intelligence (AI), its foundations and applications.
2. Apply basic principles of AI techniques in solutions that require problem solving.
3. Inference, perception, knowledge representation, and learning.
4. Use propositional and predicate logic to solve problems in AI.
5. Demonstrate proficiency developing applications in an AI programming language.



### Text Books:

1. David W. Rolston, “**Principles of Artificial Intelligence and Expert System Development**”, McGraw Hill, 2019.
2. Elaine Rich, Kevin Knight: “**Artificial Intelligence**”, Tata McGraw Hill, 2019.
3. D.W. Patterson, “**Introduction to AI and Expert Systems**”, PHI, 2018.
4. Russell, S., Norvig, P., “**Artificial Intelligence: A Modern Approach**”. Create Space Independent Publishing Platform, 2016.

### Reference Books:

1. W.F. Clocksin and Mellish, “**Programming in PROLOG**”, Narosa Publishing edition, 1991. House, 3<sup>rd</sup> Edition, 2001.
2. Ivan Bratko, “**Prolog Programming for Artificial Intelligence**”, Addison-Wesley, Pearson Education, 3<sup>rd</sup> Edition, 2000.

<b>Semester</b>	<b>: VII</b>
<b>Course Type</b>	<b>: DSC</b>
<b>Course Code</b>	<b>: CSCDSC404</b>
<b>Name of the Course</b>	<b>: Lab on DACA &amp; Compiler Design</b>
<b>Learning level</b>	<b>: Advanced Course</b>
<b>Credits</b>	<b>: 4</b>
<b>Contact Hours</b>	<b>: 60</b>
<b>Total Marks</b>	<b>: 100</b>
<b>End Semester Marks</b>	<b>: 70</b>
<b>Internal Marks</b>	<b>: 30</b>

*Course Objectives: The course objective is to*

The objective of laboratory paper on Design and Analysis of Computer Algorithms and Compiler Design is to learn implementation of different algorithms to understand the running time complexity practically. Students will execute different algorithms of incremental methods, divide and conquer methods, dynamic programming, backtracking, heuristic approach and evaluate the effectiveness. Students will also get the opportunity about designing various components of compiler.

*This paper provides practical knowledge of DACA and Compiler Design. List of laboratory programming assignments (not limited to these):*

### Lab on DACA

Problems related to Design and Analysis of Computer Algorithms should be solved by using the Programming languages C/C++/JAVA (preferably on Unix/Linux/Solaris operating systems environment on a network).

1. Stack and queues, tree, heap and heap sort, graphs and hashing.
2. Divide and conquer method: binary search, merge sort, quick sort, matrix multiplication, minimum spanning tree.
3. Dynamic programming: multistage graphs, all pair shortest paths, optimal binary search trees, traveling salesperson problem, flow shop scheduling.
4. Search and traversal techniques: AND/OR graphs, game trees, bi connected components and depth search.

5. Backtracking: Hamilton cycles, the fast Fourier transform, NP-HARD and NP complete problems.

### **Lab on Compiler Design**

Problems related to Compiler Design should be solved by using the Programming languages C/C++/JAVA as well as various tools for Compiler Construction and Design like LEX, YACC etc.

1. Construction of a lexical analyzer and LL(1) parser for a subset of FORTRAN/PASCAL/C/C++ (to be done without using any generator).
2. Construction of a lexical analyzer and LALR(1)/LR(1) parser for a subset of C/C++ (generators like LEX, YACC to be used).
3. A construction of a translator from a high level to an intermediate language which is also a very simple subset of C (The correctness of this translation may be checked by compiling this intermediate program by a standard compiler).
4. Construction of a target code generator from the above intermediate language program to the assembly language of a suitable target machine (e. g. Intel 8088). Addition of rudimentary code optimization (like peep-hole)/jump optimization.
2. Register optimization to the generated compiler.
3. Experiments with incorporation of debugging features.

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

1. Implement different data structures and analyze the running time.
2. Understand the effectiveness of divide and conquer methods by implementing different standard algorithms.
3. Compare the performance of Dynamic programming with the divide and conquer method.
4. Design different algorithms for NP-HARD and NP complete problems.
5. Design various components of compiler

<b>Semester</b>	<b>: VIII</b>
<b>Course Type</b>	<b>: DSC</b>
<b>Course Code</b>	<b>: CSCDSC451</b>
<b>Name of the Course</b>	<b>: Software Engineering</b>
<b>Learning level</b>	<b>: Advanced Course</b>
<b>Credits</b>	<b>: 4</b>
<b>Contact Hours</b>	<b>: 60</b>
<b>Total Marks</b>	<b>: 100</b>
<b>End Semester Marks</b>	<b>: 70</b>
<b>Internal Marks</b>	<b>: 30</b>

**Course Objectives:** *The course objective is to*

1. Understand the concept of software engineering
2. Introduce the various stages of Software Development Life Cycle
3. Introduce the various process models.
4. Introduce the software quality, reliability and configuration management
5. Learn the design & testing principles to software project development

### **Unit-I**

**Introduction:** Importance of software, Software Characteristics, importance of Software engineering, Software Process Framework. Life Cycle Models: Waterfall model, Iterative models, RAD Model, Prototyping Model, Spiral Model.

### **Unit-II**

**Requirement Analysis:** Software Requirement Analysis, Initiating Requirement Engineering Process, Requirement Analysis and Modeling Techniques, Flow Oriented Modeling, Need for SRS, Characteristics and Components of SRS. **Software Project Management:** Estimation in Project Planning Process, Project Scheduling.

### **Unit-III**

**Risk Management:** Software Risks, Risk Identification, Risk Projection and Risk Refinement, RMMM Plan. **Quality Management:** Quality Concepts, Software Quality Assurance, Software Reviews, Metrics for Process and Projects.

### **Unit-IV**

**Software Design:** Conceptual and Technical Designs, Objectives of software Design, Modularity: Module Coupling, Module Cohesion, Relation between cohesion and coupling. Design Strategy: Bottom-up, Top-Down. Function oriented design: design notations, Functional Procedural layers, DFD, Flowchart, Structure charts, Transform and transaction analysis. Object Oriented Design: Basic mechanism, concepts, advantages of OOD, unified modeling language (UML).

### **Unit-V**

**Testing Strategies & Tactics:** Software Testing Fundamentals, Strategic Approach to Software Testing, Unit testing, Integration testing, Validation Testing, System testing, Black-Box Testing, White-Box Testing and their type, Basis Path Testing.

**Course Learning Outcomes:** *After successful completion of the course, the students will be able to:*

- 1. Understand the various stages of SDLC, the various process models, software quality, reliability and configuration management.*
- 2. Explain SRS documents, planning, analysis, design and development of software*
- 3. Implement, test and validate a system designs and develop the software*

### **Text Books:**

1. Roger S. Pressman: **Software Engineering, A Practitioner's Approach**, 7<sup>th</sup> Edition, Tata McGraw Hill.
2. Rajib Mall: **Fundamentals of Software Engineering**, 5<sup>th</sup> Edition, PHI

### **Reference Books:**

1. Pankaj Jalote, **An Integrated Approach to Software Engineering**, Narosa Publishing House 2011, 3<sup>rd</sup> Edition.
2. K. K. Aggarwal, Yogesh Singh: **Software engineering**, 3<sup>rd</sup> Edition, New Age International publishers
3. F. Tsui, O. Karam, B. Bernal: **Essentials of Software Engineering**, 4<sup>th</sup> Edition, Jones & Bartlett

<b>Semester</b>	<b>: VIII</b>
<b>Course Type</b>	<b>: DSC</b>
<b>Course Code</b>	<b>: CSCDSC452(A)</b>
<b>Name of the Course</b>	<b>: Image Processing</b>
<b>Learning level</b>	<b>: Advanced Course</b>
<b>Credits</b>	<b>: 4</b>
<b>Contact Hours</b>	<b>: 60</b>
<b>Total Marks</b>	<b>: 100</b>
<b>End Semester Marks</b>	<b>: 70</b>
<b>Internal Marks</b>	<b>: 30</b>

*Course Objectives: The course objective is to*

1. Introduce the concepts of Digital Image Processing and fundamental techniques
2. Explain the concepts of Digital Image Transforms
3. Understand image compression techniques
4. Explain Image segmentation techniques

### **Unit-I**

**Digital image fundamentals** - Digital Image representation, Image acquisition, Fundamental steps in Image processing, Elements of digital Image processing systems, Types of Images, Gray level to binary image conversion, Sampling and quantization

### **Unit-II**

**Image Transforms:** Need for Image transform, Fourier Transform, 2D Discrete Fourier Transform, Properties of 2D FFT, Walsh transform, Hadamard Transform, Discrete cosine Transform, Haar transform, Slant transform, Hotelling transform

### **Unit-III**

**Image enhancement:** Point processing. Histogram processing. Spatial filtering. Enhancement in frequency domain, Image smoothing, Image sharpening.

### **Unit-IV**

**Image Compression:** Image Compression Models, Source encoder and decoder, Error free compression, Lossy compression, Image Compression Measures, Huffman Coding.

### **Unit-V**

**Image segmentation:** Detection of discontinuities, Edge linking and boundary detection, Thresholding, Region oriented segmentation.

***Course Learning Outcomes:** After successful completion of the course, the students will be able to:*

1. Learn image conversion and image transformation techniques
2. Apply image enhancement techniques
3. Learn image compression models.
4. Apply various image processing techniques for real time applications

### **Text Books:**

1. R. C. Gonzalez, R. E. Woods, “**Digital Image Processing**”, 4<sup>th</sup> Edition, Pearson Education, 2018.
2. A. K. Jain, “**Fundamentals of Digital Image Processing**”, Prentice Hall of India.

1989.

**Reference Books:**

1. K. R. Castleman, “**Digital Image Processing**”, Pearson Education, 1996.
2. Rafael C. Gonzalez, Richard E. Woods, Steve Eddins, “**Digital Image Processing Using MATLAB**”, Pearson Education, Inc., 2004.
3. William K. Pratt, **Digital Image Processing**, John Wiley, 3<sup>rd</sup> Edition, 2004.

<b>Semester</b>	<b>: VIII</b>
<b>Course Type</b>	<b>: DSC</b>
<b>Course Code</b>	<b>: CSCDSC452 (B)</b>
<b>Name of the Course</b>	<b>: Data Analytics</b>
<b>Learning level</b>	<b>: Advanced Course</b>
<b>Credits</b>	<b>: 4</b>
<b>Contact Hours</b>	<b>: 60</b>
<b>Total Marks</b>	<b>: 100</b>
<b>End Semester Marks</b>	<b>: 70</b>
<b>Internal Marks</b>	<b>: 30</b>

*Course Objectives: The course objective is to*

1. *Explain the basics of data science and its applications*
2. *Learn the strategies of data collection and pre-processing*
3. *Explain data analytics using Python*
4. *Learn the application of AI in data analytics*

### **Unit-I**

**Introduction** - Introduction to Data Analytics – Evolution of Data Analytics – Data Analytics Roles – Stages in a Data Analytics Project – Applications of Analytics in various fields – Data Security Issues. Data Collection Strategies – Data Pre-Processing Overview – Data Cleaning – Data Integration and Transformation – Data Reduction – Data Discretization.

### **Unit-II**

**Deep Dive with Excel:** Data Connectors in Excel, Cleaning in Power Query Editor, Adding Conditional Columns using Power Query, Editor, Data Modelling and its Importance, Cardinality and Filter Direction in Power Pivot, Pivot Tables in Excel, Charts in Excel, Creating a Dashboard in Excel

### **Unit-III**

**Python for Data Analytics:** Introduction to Python for Data Analytics, In-built Modules, Pickle Library, Introduction to Numpy, Statistical Functions in Arrays, WebScraping, BeautifulSoup and Requests Library in Python, Extracting Data from Tables, Extracting Data from Multi-Page Websites, Text Analysis using Python.

### **Unit-IV**

**Machine Learning: Introduction,** Applications of Machine Learning, Linear Regression, Decision Trees, Random Forests and Ensemble Methods, Clustering Algorithms, Dimensionality Reduction.

### **Unit-V**

**AI for Data Analytics:** Generating code & debugging errors, Exchanging Data Cleaning Skills, Crafting SQL queries, Optimizing Model Selection with AI-driven Methods, Utilizing AI for Anomaly Detection in Data Analysis

**Course Learning Outcomes:** *After successful completion of the course, the students will be able to:*

1. *Understand the concept of data science and applications*
2. *Understand the methods of data collection and processing*
3. *Perform data analysis using python*
4. *Employ AI for data analytics*

**Text Books:**

1. Cathy O’Neil and Rachel Schutt , “**Doing Data Science**”, O'Reilly, 2015.
2. Fabio Nelli, “**Python Data Analytics**”, Apress, Publishing, 1st Edition, 2015

**Reference Books:**

1. G. Sudhamathy and C. Jothi Venkateswaran, **R Programming: An Approach to Data Analytics**, MJP Publishers, 2019

<b>Semester</b>	<b>: VIII</b>
<b>Course Type</b>	<b>: DSC</b>
<b>Course Code</b>	<b>: CSCDSC453</b>
<b>Name of the Course</b>	<b>: Natural Language Processing</b>
<b>Learning level</b>	<b>: Advanced Course</b>
<b>Credits</b>	<b>: 4</b>
<b>Contact Hours</b>	<b>: 60</b>
<b>Total Marks</b>	<b>: 100</b>
<b>End Semester Marks</b>	<b>: 70</b>
<b>Internal Marks</b>	<b>: 30</b>

*Course Objectives: The course objective is to*

1. *Learn the basics of human language structure and computational linguistics*
2. *Understand the core concepts and challenges of NLP*
3. *Understand Morphological Analysis part-of-speech tagging and syntactic parsing*
4. *Work with machine translation, speech recognition, and text-to-speech systems.*

**Unit I**

**Introduction to NLP:** Some example application, Achievements and brief history, Open problems, Major Goals. Introduction to Language Structure and Language Analyzer: Introduction to Language Structure, Overview of Language Analyzer Requirement of Computational Grammars.

**Unit II**

**Linguistics Resources:** Introduction to corpus, elements in balanced corpus, TreeBank, PropBank. **Pre-processor:** Objectives of Processor, Analyzing a text, Discourse Language, Punctuation’s, Abbreviations, names, special characters, Need for human Preprocessor.

**Unit III**

**Words and their Analyzer :** Introduction, Why Morphological Analysis, Morphological Generation using paradigms, Morphological Analysis using Paradigms, Speedup Morphological Analysis by compilation, Morphological Analyzer – Some Additional Issues.

#### Unit IV

**Part of Speech tagging-** Stochastic POS tagging, Hidden Markov Models (HMM), Transformation based tagging (TBL), Handling of unknown words, named entities, multi word expressions.**Rule Base:** Sentences and their analysis, Why Rule Base, Application in Rule Base, Verb Groups, Noun Group, Strategy for Grammar Development, and Semantics in stages.

#### Unit V

**Parsing-** Unification, probabilistic parsing, TreeBank. **Semantics-** Meaning representation, semantic analysis, lexical semantics, WordNet.**Machine Translation:** Introduction, Problems of Machine Translation, Brief History, Possible Approaches, ANGLABHARTI and its importance. Applications of NLP- Spell-checking, Summarization.

**Course Learning Outcomes:** After successful completion of the course, the students will be able to:

1. Explain the fundamental principles and challenges of NLP.
2. Describe the structure of human language and its computational representation
3. Perform part-of-speech tagging, named entity recognition (NER), and syntactic parsing
4. Build real-world NLP applications such as chatbots, sentiment analysis, text summarization, and machine translation

#### Textbook:

1. Daniel Jurafsky and James H Martin. Speech and Language Processing, 2<sup>nd</sup> Edition, Pearson Education, 2009

#### Reference Books:

1. James A.. **Natural language Understanding**, Pearson Education, 2<sup>nd</sup> Edition, 1994
2. Bharati A., Sangal R., ChaitanyaV.. Natural language processing: a Paninian perspective, PHI, 2000
3. Siddiqui T., Tiwary U. S.. Natural language processing and Information retrieval, OUP, 2008

<b>Semester</b>	<b>: VIII</b>
<b>Course Type</b>	<b>: DSC</b>
<b>Course Code</b>	<b>: CSCDSC454(A)</b>
<b>Name of the Course</b>	<b>: Internet of Things</b>
<b>Learning level</b>	<b>: Advanced Course</b>
<b>Credits</b>	<b>: 4</b>
<b>Contact Hours</b>	<b>: 60</b>
<b>Total Marks</b>	<b>: 100</b>
<b>End Semester Marks</b>	<b>: 70</b>
<b>Internal Marks</b>	<b>: 30</b>

**Course Objectives:** The course objective is to

1. Understand the basic concepts of IoT,
2. Understand the application areas of IoT.
3. Explain the concepts of IoT devices and sensors.
4. To understand the future trends of IoT.

#### Unit-I

**Introduction to IoT:** Definition and Concept of IoT, Evolution and Importance of IoT, Components of IoT Ecosystem, IoT Architecture and Communication Protocols, IoT Applications in Different Industries, Ethical and Privacy Considerations in IoT

## **Unit-II**

**IoT Devices and Sensors:** Types of IoT Devices (Sensors, Actuators, Smart Objects), IoT Device Connectivity (Wired, Wireless, Cellular), IoT Access Technologies, Data Acquisition and Sensor Fusion, Applications protocols for IoT

## **Unit-III**

**Data and Analytics for IoT:** An introduction to data analytics for IoT, Machine Learning, Big data analytics tools and technology, edge streaming analytics, network analytics.

## **Unit-IV**

**IoT Networking and Security Networking Models for IoT** (Star, Mesh, etc.), IPv6 and IP Addressing in IoT, Wireless Sensor Networks, Security Challenges in IoT, IoT Security Measures (Authentication, Encryption, etc.), IoT Security Best Practices and Standards

## **Unit-V**

**IoT Applications and Future Trends** Smart Cities and Urban IoT, Industrial IoT (IIoT) and Industry 4.0, Healthcare and Wearable IoT, Agriculture and Environmental Monitoring, Home Automation and Consumer IoT, Emerging Trends and Future Directions of IoT.

**Course Learning Outcomes:** *After successful completion of the course, the students will be able to:*

1. *Explain the Architecture of IoT.*
2. *Understand different types of sensor and IoT network.*
3. *Learn the applications of IoT and its future*

### **Text Books:**

1. D. Hanes, G. Salgueiro, P. Grossetete, R. Barton, J. Henry, **IoT Fundamentals: Networking Technologies, Protocols, and Use Cases for the Internet of Things**, Cisco Press, 2017.
2. Raj Kamal, **Internet of Things (Architecture and Design Principles)**, McGraw Hill Education, 2017.

### **Reference Books:**

1. Arshdeep Bahga and Vijay Madisetti, **"Internet of Things: A Hands-On Approach"**, Orient Blackswan Private Limited, 2015
2. Cuno Pfister **"Getting Started with the Internet of Things"** O'reilly Media, INC International Concepts USA, 2011

<b>Semester</b>	<b>: VIII</b>
<b>Course Type</b>	<b>: DSC</b>
<b>Course Code</b>	<b>: CADSC454 (B)</b>
<b>Name of the Course</b>	<b>: Cloud Computing</b>
<b>Learning level</b>	<b>: Advanced Course</b>
<b>Credits</b>	<b>: 4</b>
<b>Contact Hours</b>	<b>: 60</b>
<b>Total Marks</b>	<b>: 100</b>
<b>End Semester Marks</b>	<b>: 70</b>



*Course Objectives: The course objective is to*

1. *Compare the strengths and limitations of cloud computing*
2. *Identify the architecture, infrastructure and delivery models of cloud computing*
3. *Apply suitable virtualization concept.*
4. *Choose the appropriate cloud player, Programming Models and approach.*
5. *Address the core issues of cloud computing such as security, privacy and interoperability*

### **Unit I**

**Introduction to Cloud Computing:** Evolution and History of Cloud Computing, Introduction to Cloud Computing, Why Cloud Computing is Becoming Highly Important, Features of Cloud Computing, Cloud Computing for various users, Advantages of Cloud Computing, Limitations of Cloud Computing.

### **Unit II**

**Cloud Models and Types:** The NIST Model, Cloud Cube Model, Deployment Models, Service Models. Layers and Types of Cloud, Components of Cloud Computing, Cloud Computing Service Providers

**Software as a Service (SaaS):** Software as a Service, Evolution of SaaS, Brief Introductory part of Software as a Service, SaaS Unification Technologies, SaaS Integration Products and Technologies, SaaS Product Selection Criteria, SaaS Integration Services, Advantages of SaaS.

### **Unit III**

**Platform as a Service (PaaS):** Introduction to PaaS, Evolution of PaaS, PaaS Service Providers- Acquia Cloud, Amazon AWS, Amazon Elastic Beanstalk, Google App Engine, Force.com, PaaS Application Framework, PaaS Operator Verbs, PaaS Developer Verbs, Advantages and Challenges of PaaS.

### **Unit IV**

**Infrastructure as a Service (IaaS):** Evolution, IaaS Architecture- Advantages and Disadvantages of Infrastructure as a Service, SAN model, IaaS Providers, IaaS Architecture, Advantages and Disadvantages of Infrastructure as a Service.

**Data in Cloud :** Evolution of Network Storage in Cloud, Data as a Service, Database as a Service, Cloud Based Data Storage, Advantages and Limitations of Cloud Based Storage Solution, Cloud Based Data Storage Service Providers.

### **Unit V**

**Virtualization:** Introduction to Virtualization and its Technical Evolution, History of Virtualization, Types of Virtual Machines, Advantages of Virtualization, Components of Virtualization, Types of Virtualization.

**Course Learning Outcomes:** *After successful completion of the course, the students will be able to:*

1. *Explain the core concepts, benefits, and challenges of cloud computing.*
2. *Describe different cloud service models (IaaS, PaaS, SaaS) and deployment models*
3. *Understand cloud architecture, data centers, and virtualization technologies*

### Text Books:

1. Dr.Anand Nayyar, **Handbook of Cloud Computing** , 1<sup>st</sup> Edition, BPB Publication, India, 2019.
2. Anthony T.Velte, Toby J. Velte Robert Elsenpeter, **Cloud computing:A practical approach** - TATA McGraw- Hill, 2010.
3. Michael Miller, **Cloud Computing: Web-Based Applications That Change the Way You Work and Collaborate Online**, Que Publication, 2008

### Reference Books:

1. Dan C Marinescu, **Cloud Computing, Theory and Practice**, MK Publisher, Import 4, 2013.
2. Arshadeep Bahga, Vijay Madisetti, **Cloud Computing, A Hands on approach**, Vijay Madisetti Publisher, 2014.
3. Raj Kumar Buyya, Christenvecctiola, S Tammaraiselvi, **Mastering Cloud Computing, Foundations and Application Programming**, TMH, 2011.

<b>Semester</b>	<b>: VIII</b>
<b>Course Type</b>	<b>: DSC</b>
<b>Course Code</b>	<b>: CSCDSC451</b>
<b>Name of the Course</b>	<b>: Research Methodology</b>
<b>Learning level</b>	<b>: Advanced Course</b>
<b>Credits</b>	<b>: 4</b>
<b>Contact Hours</b>	<b>: 60</b>
<b>Total Marks</b>	<b>: 100</b>
<b>End Semester Marks</b>	<b>: 70</b>
<b>Internal Marks</b>	<b>: 30</b>
<b>(***To be prepared at the university level)</b>	

<b>Semester</b>	<b>: VIII</b>
<b>Course Type</b>	<b>: DSC</b>
<b>Course Code</b>	<b>: CSCDSC455</b>
<b>Name of the Course</b>	<b>: Research Project/Dissertation</b>
<b>Learning level</b>	<b>: Advanced Course</b>
<b>Credits</b>	<b>: 12</b>
<b>Contact Hours</b>	<b>: 360</b>
<b>Total Marks</b>	<b>: 300</b>
<b>End Semester Marks</b>	<b>: 210</b>
<b>Internal Marks</b>	<b>: 90</b>

- ✓ *Honours students not undertaking research will do 3 courses for 12 credits in lieu of a research project / Dissertation.*
- ✓ Students who secure 75% marks and above in the first six semesters and wish to undertake research at the undergraduate level can choose a research stream in the fourth year. They should do a research project or dissertation under the guidance of a faculty member of the University/College. The research project/dissertation will be in the major discipline. The students who secure 160 credits, including 12 credits from a research project/dissertation , are awarded UG Degree (Honours with Research).

- ✓ The research outcomes of their project work may be published in peer-reviewed journals or may be presented in conferences /seminars or may be patented.
- ✓ The final semester will be devoted to seminar presentation, preparation, and submission of project report/dissertation. The project work/dissertation will be on a topic in the disciplinary programme of study or an interdisciplinary topic.
- ✓ Students choose a research component with courses relating to research methodology, advanced courses in theory and applied areas, and seminar presentations. Students may be permitted to carry out a research project or dissertation in another department of the same institution or another institution provided the required facilities are available.

<b>Semester</b>	<b>: VII</b>
<b>Course Type</b>	<b>: DSM</b>
<b>Course Code</b>	<b>: CSCDSM401</b>
<b>Name of the Course</b>	<b>: Internet Technology</b>
<b>Learning level</b>	<b>: Advanced Course</b>
<b>Credits</b>	<b>: 4</b>
<b>Contact Hours</b>	<b>: 60</b>
<b>Total Marks</b>	<b>: 100</b>
<b>End Semester Marks</b>	<b>: 70</b>
<b>Internal Marks</b>	<b>: 30</b>

*Course Objectives: The course objective is to*

1. *Understand basic Internet architecture, protocols, and standards*
2. *Explore networking concepts like TCP/IP, HTTP, FTP, and DNS*
3. *Learn about web development basics: HTML, CSS, JavaScript and PHP.*

## **UNIT I**

**Internet, structure of internet, history of internet, Internet protocol:** TCP/IP, SLIP, PPP, Network and network devices, Addressing in Internet - DNS, domain name and their organisation, understanding the Internet protocol address, Client-server concept- architecture and application. Evolution of www, basic features, servers, http, FTP, URL, search engine, searching categories, hypertext.

## **UNIT II**

**HTML:** Basic HTML, HTML tags, creating list in HTML, hyperlinks, multimedia, HTML forms, tables in HTML, frames in HTML, image maps, style sheets in HTML. DHTML, XML-Introduction, Need for XML, Advantages, simple XML programs, DTD.

## **UNIT III**

**Creating interactive and dynamic web pages with JavaScript:** Client-side scripting languages, JavaScript overview; constants, variables, operators, expressions and statements; user-defined & built-in-functions; properties and methods of built-in objects, client-side form validation.

## **UNIT IV**

**Introduction to PHP:** Server side scripting language, Evaluation of PHP, Basic Syntax, Defining variable and constant, PHP Data type, Operator and Expression. Decisions and loop Making Decisions, Doing Repetitive task with looping, Mixing Decisions and looping with

## HTML

### UNIT V

**Handling Html Form with PHP:** Capturing Form, Data Dealing with Multi-value filed, and generating File uploaded form, redirecting a form after submission. Database Connectivity with MySQL: Introduction to RDBMS, Connection with MySQL Database, Performing basic database operation.

*Course Learning Outcomes: After successful completion of the course, the students will be able to:*

1. *Explain the history, evolution, and architecture of the Internet*
2. *Develop basic web pages using HTML, CSS, and JavaScript.*
3. *Understand how web servers, browsers, and databases interact.*

#### Text Books:

1. M.L Young, **The Internet: Complete**; Tata McGraw Hill, 3<sup>rd</sup> Edition, 2006.
2. J. Jaworski, **Mastering JavaScript and Jscript**, BPB Publication, 2<sup>nd</sup> Edition, 2011.
3. D. Godman, **Dynamic HTML: The definitive references**, Shroff Publishers, 2<sup>nd</sup> Edition, 2009.
4. Steven Holzner, **PHP: The Complete Reference Paperback**, McGraw Hill Education (India), 2007.

#### Reference Books:

1. Daniel Minoli, **Internet and Internet Engineering**, Tata McGraw-Hill Edition, 2<sup>nd</sup> Edition, 2011.
2. Luke Welling, Laura Thompson, **PHP and MySQL Web Development**, 4<sup>th</sup> Edition, Addition Paperback, Addison-Wesley Professional, 2008.

<b>Semester</b>	<b>: VIII</b>
<b>Course Type</b>	<b>: DSM</b>
<b>Course Code</b>	<b>: CSCDSM451</b>
<b>Name of the Course</b>	<b>: Internet Technology</b>
<b>Learning level</b>	<b>: Advanced Course</b>
<b>Credits</b>	<b>: 4</b>
<b>Contact Hours</b>	<b>: 60</b>
<b>Total Marks</b>	<b>: 100</b>
<b>End Semester Marks</b>	<b>: 70</b>
<b>Internal Marks</b>	<b>: 30</b>

*Course Objectives: The course objective is to*

1. *Understand basic Internet architecture, protocols, and standards*
2. *Explore networking concepts like TCP/IP, HTTP, FTP, and DNS*
3. *Learn about web development basics: HTML, CSS, JavaScript and PHP.*

### UNIT I

**Internet, structure of internet, history of internet, Internet protocol:** TCP/IP, SLIP, PPP, Network and network devices, Addressing in Internet - DNS, domain name and their organisation, understanding the Internet protocol address, Client-server concept- architecture and application. Evolution of www, basic features, servers, http, FTP, URL, search engine,

searching categories, hypertext.

## UNIT II

**HTML:** Basic HTML, HTML tags, creating list in HTML, hyperlinks, multimedia, HTML forms, tables in HTML, frames in HTML, image maps, style sheets in HTML. DHTML, XML-Introduction, Need for XML, Advantages, simple XML programs, DTD.

## UNIT III

**Creating interactive and dynamic web pages with JavaScript:** Client-side scripting languages, JavaScript overview; constants, variables, operators, expressions and statements; user-defined & built-in-functions; properties and methods of built-in objects, client-side form validation.

## UNIT IV

**Introduction to PHP:** Server side scripting language, Evaluation of PHP, Basic Syntax, Defining variable and constant, PHP Data type, Operator and Expression. Decisions and loop Making Decisions, Doing Repetitive task with looping, Mixing Decisions and looping with HTML

## UNIT V

**Handling Html Form with PHP:** Capturing Form, Data Dealing with Multi-value filed, and generating File uploaded form, redirecting a form after submission. Database Connectivity with MySQL: Introduction to RDBMS, Connection with MySQL Database, Performing basic database operation.

*Course Learning Outcomes: After successful completion of the course, the students will be able to:*

1. *Explain the history, evolution, and architecture of the Internet*
2. *Develop basic web pages using HTML, CSS, and JavaScript.*
3. *Understand how web servers, browsers, and databases interact.*

### Text Books:

1. M.L Young , **The Internet: Complete**; Tata McGraw Hill, 3<sup>rd</sup> Edition, 2006.
2. J. Jaworski, **Mastering JavaScript and Jscript**, BPB Publication, 2<sup>nd</sup> Edition, 2011.
3. D. Godman, **Dynamic HTML: The definitive references**, Shroff Publishers, 2<sup>nd</sup> Edition, 2009.
4. Steven Holzner, **PHP: The Complete Reference Paperback**, McGraw Hill Education (India), 2007.

### Reference Books:

1. Daniel Minoli, **Internet and Internet Engineering**, Tata McGraw-Hill Edition, 2<sup>nd</sup> Ed., 2011.
2. Luke Welling, Laura Thompson, **PHP and MySQL Web Development**, 4<sup>th</sup> Edition, Addition Paperback, Addison-Wesley Professional, 2008.