Department of Computer Science

Bhairab Ganguly College

Course Outcomes – B.Sc. Computer Science

B.Sc. in Computer Science program enables students to attain the ability to:

- Analyze a problem, and identify and define the computing requirements appropriate to its solution
- Design, implement, and evaluate a computer-based system, process, component, or program to meet desired needs
- Work effectively on teams to accomplish a common goal
- Understand the professional, ethical, legal, security and social issues and responsibilities
- Analyze the local and global impact of computing on individuals, organizations, and society
- Recognize the need for and an ability to engage in continuing professional development
- Implement current techniques, skills, and tools necessary for computing practice.
- Apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer-based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices.
- Apply design and development principles in the construction of software systems of varying complexity.

Course Specific Outcomes – B.Sc. Computer Science (HONOURS)

SEMESTER-I

CMSACOR01T: Programming Fundamentals using C/C++

On the completion of the course, students acquire theoretical as well as practical knowledge of:

- Procedural Programming and Object-Orientation Programming
- Data Types, Variables, Constants, Operators and Basic I/O
- Expressions, Conditional Statements and Iterative Statements
- Functions and Arrays
- Derived Data Types (Structures and Unions)
- Pointers and References in C++

- Memory Allocation in C++
- File I/O, Preprocessor Directives
- Classes in C++
- Function Overloading and Operator Overloading
- Inheritance, Polymorphism and Exception Handling

CMSACOR02T: Computer System Architecture

On the completion of the course, students have the fundamental knowledge of:

- Logic gates, boolean algebra, combinational circuits and sequential circuits
- Computer Arithmetic
- Computer Organization and Design
- Central Processing Unit
- Memory Organization
- Input-Output Organization

SEMESTER-II

CMSACOR03T: Programming in Java

On the completion of the course, students acquire theoretical as well as practical knowledge of:

- Object-Oriented Programming
- Java Architecture and Features
- Semantic and syntax differences between C++ and Java
- Inheritance, Interfaces, Packages, Enumerations, Autoboxing and Metadata
- Exception Handling, Threading, Networking and Database Connectivity
- Applets and Event Handling

CMSACOR04T: Discrete Structures

On the completion of the course, students acquire the fundamental knowledge of:

- Set, Relation and function
- Permutation and Combination
- Mathematical Induction
- Principle of Inclusion and Exclusion.
- Asymptotic Notations

- Recurrence Relations
- Graph Theory
- Prepositional Logic

SEMESTER-III

CMSACOR05T: Data Structures

On the completion of the course, students acquire theoretical as well as practical knowledge of:

- Arrays and its Representation by Sparse Matrices
- Stacks-Implementation and application
- Linked Lists Data Structure
- Queues Data Structure
- Recursion
- Tree as a data structure, Binary Trees, Binary Search Trees, Threaded Binary Trees, and Height-Balanced Trees
- Searching and Sorting
- Hashing

CMSACOR06T: Operating Systems

On the completion of the course, students have the fundamental knowledge of:

- Functions and types of operating systems
- Operating System Organization which includes the study of Processor and user modes, kernels, system calls and system programs.
- Process Management that includes threads, thread libraries, Process Scheduling, scheduling algorithms, critical section, semaphores and deadlocks.
- Memory Management that includes Physical address space, virtual address space, memory allocation strategies, paging, segmentation and virtual memory.
- File and I/O Management
- Protection and Security mechanism.

CMSACOR07T: Computer Networks

On the completion of the course, students have the fundamental knowledge of:

- Computer Networks, network topologies, network classifications, OSI reference model and TCP/IP protocol suite.
- Analog and digital signal, parallel and serial transmission, digital to analog modulation, pulse code modulation; multiplexing techniques- FDM, TDM and transmission media.
- Networks Switching Techniques

- Error detection and error correction techniques
- flow control and error recovery protocols- stop and wait ARQ, go-back-n ARQ.
- Multiple Access Protocol-CSMA/CD and CSMA/CA protocols.
- Ethernet LANS.
- Networking Devices-repeaters, hubs, switches, bridges, router and gateways.
- Routing algorithms
- IP protocol and Internet control protocols.
- Error and flow control in Transport Layer, three-way handshake;
- DNS, WWW & HTTP protocol.

SEMESTER-IV

CMSACOR08T: Design and Analysis of Algorithms

On the completion of the course, students acquire both theoretical and practical knowledge of:

- Basic Design and Analysis techniques of Algorithms.
- Iterative techniques, Divide and Conquer, Dynamic Programming and Greedy Algorithms.
- Sorting and Searching Techniques-Bubble Sort, Insertion Sort, Merge Sort, Heap Sort, Quick Sort, Bucket Sort, Radix Sort and Count Sort.
- Complexity analysis of Algorithms.
- Decision Trees and Balanced Trees (Red-Black Trees)
- Amortized analysis
- Graph Algorithms–Breadth First Search, Depth First Search and its Applications
- Minimum Spanning Trees.
- String Processing-String Matching and KMP Technique.

CMSACOR09T: Software Engineering

On the completion of the course, students acquire the knowledge of:

- Software Engineering Technology and Process Models.
- Software Requirement Analysis
- Characteristics and Components of SRS
- Software Project Management-Estimation in Project Planning Process and Project Scheduling.
- Software Risk Management-Risk Identification, Projection and Refinement, RMMM Plan.
- Software Quality Management- Software Quality Assurance, Software Reviews and Metrics for Process and projects.
- Design Engineering-Software Architecture and Modeling component level design.
- Software Testing Strategies & Tactics-Validation Testing, System testing, Black-Box Testing, White-Box Testing and Basis Path Testing.

CMSACOR10T: Database Management Systems

On the completion of the course, students acquire both theoretical and practical knowledge of:

- Database system architecture
- Entity Relationship (ER) Modeling
- Relational model concepts and relational constraints
- Relational algebra
- SQL queries
- Functional Dependencies
- Lossless decomposition
- Normalization (upto BCNF).
- ACID properties and concurrency control
- File Structure and Indexing-Primary index, secondary index, clustering index and Multilevel indexing using B and B+ trees.

SEMESTER-V

CMSACOR11T: Internet Technologies

On the completion of the course, students acquire both theoretical and practical knowledge of:

- JavaScript-Data types, operators, functions, control structures, events and event handling.
- JDBC Fundamentals, Establishing Connectivity and working with connection interface.
- Creating and Executing SQL Statements.
- Introduction to JavaServer Pages, HTTP and Servlet Basics.
- JSP-Anatomy of a JSP Page, JSP Processing, JSP Application Design with MVC.
- Error Handling and Debugging.
- Java Beans Fundamentals, JAR files, Introspection, Developing a simple Bean and Connecting to DB

CMSACOR12T: Theory of Computation

On the completion of the course, students acquire the fundamental knowledge of:

- Deterministic and non-deterministic finite automata (DFA & NFA)
- Regular languages and their relationship with finite automata
- Context free grammars
- Parse trees, ambiguities in grammars and languages
- Pushdown automata
- Pumping Lemma
- Properties of context free languages
- Turing Machine as a model of computation
- Language acceptability, decidability and halting problem
- Recursively enumerable and recursive languages

- unsolvability problems.

CSMADSE01T: Microprocessor

On the completion of the course, students acquire the fundamental knowledge of:

- Microprocessor architecture-Internal architecture, system bus architecture, memory and I/O interfaces.
- Register Organization
- Microprocessor programming-Instruction formats and assembly language programming.
- Memory address decoding
- cache memory and cache controllers
- I/O interface, keyboard, display and timer.
- DMA controller, interrupt controller, video controllers and communication interfaces.

CSMADSE03T: Cloud Computing

On the completion of the course, students acquire the knowledge of:

- Recent trends in Computing-Grid Computing, Cluster Computing, Distributed
- Computing, Utility Computing and Cloud Computing,
- Cloud Computing fundamentals, Cloud service providers, Benefits and limitations of Cloud Computing.
- Service Models for cloud computing-Infrastructure as a Service (IaaS), Platform as a Service (PaaS) and Software as a Service (SaaS).
- Cloud Computing Working Mechanism
- Deployment Models- Public cloud, Private cloud, Hybrid cloud and Community cloud.
- Case Studies of Cloud Computing-Google App Engine, Microsoft Azure, Amazon EC2 and Eucalyptus.
- Service Management in Cloud Computing-Service Level Agreements (SLAs), Billing & Accounting.
- Economics of scaling in Cloud Computing.
- Cloud Security-Network level security, Host level security, Application level security, Data security, Jurisdictional issues raised by Data location and Authentication in cloud computing.

SEMESTER-VI

CMSACOR13T: Artificial Intelligence

On the completion of the course, students acquire the fundamental knowledge of:

- Artificial Intelligence and its Applications

- Problem Solving and Searching Techniques-Breadth First Search, Depth First Search, Hill climbing and its Variations, Heuristics Search Techniques: Best First Search and A* algorithm.
- Constraint Satisfaction Problem, Game Playing, Min-Max and Alpha-Beta pruning algorithms.
- First Order Predicate Logic
- Semantic Nets and Programming in Logic (PROLOG)
- Dealing with Uncertainty and Inconsistencies-Truth Maintenance System, Default Reasoning, Probabilistic Reasoning, Bayesian Probabilistic Inference and Possible World Representations.
- Natural Languages Parsing Techniques, Context-Free and Transformational Grammars, Recursive and Augmented Transition Nets.

CMSACOR14T: Computer Graphics

On the completion of the course, students acquire the fundamental knowledge of:

- Computer Graphics and its Applications
- Graphics Hardware-Architecture of Raster and Random scan display devices and input/output devices.
- Fundamental Techniques in Graphics-Raster scan line, circle and ellipse drawing, thick primitives, Polygon filling, line and polygon clipping algorithms, 2D and 3D Geometric Transformations, 2D and 3D Viewing Transformations (Projections- Parallel and Perspective), Vanishing points. Geometric Modeling -Representing curves & Surfaces.
- Visible Surface determination-Hidden surface elimination.
- Surface rendering-Illumination and shading models. Basic color models and Computer Animation.

CSMADSE04T: Big Data

On the completion of the course, students acquire the knowledge of:

- Basics of big data, big data and marketing, credit risk management for big data, big data and healthcare, big data in medicine.
- NoSQL Data Management-aggregate data models, graph databases, schema less databases,
- distribution models, partitioning and combining.
- Basics of Hadoop-Data format, Hadoop streaming and Hadoop pipes
- Hadoop distributed file system (HDFS)-HDFS concepts, Java interface and Hadoop I/O
- Data integrity, compression, serialization, Avro, file-based data structures.
- MapReduce Applications-Unit tests with MRUnit and test data and anatomy of MapReduce job run
- YARN-failures in classic Map-reduce, job scheduling, shuffle and sort.
- Hbase-data model and implementations, Hbase clients and Hbase examples
- Cassandra – Cassandra data model, Cassandra examples, Cassandra clients

- Pig-pig data model, Pig Latin, developing and testing Pig Latin scripts.
- Hive-HiveQL data definition, HiveQL data manipulation and HiveQL queries.

CMSADSE05T: Digital Image Processing

On the completion of the course, students acquire the knowledge of:

- Light, Brightness adaption and discrimination, Pixels, Perspective Projection, sampling and quantization.
- Spatial Domain Filtering-Smoothing filters, sharpening filters, gradient and Laplacian.
- Hotelling Transform, Fourier Transforms and Discrete Cosine Transform.
- FFT, Convolution, Correlation, 2-D sampling and Frequency domain filtering.
- Image Restoration-Restoration techniques, Noise restoration filters, Adaptive filters, Linear, Estimation of Degradation functions and Restoration from projections.
- Image Compression-Huffman Coding, Arithmetic Coding, Golomb Coding, LZW coding, Transform Coding, Symbol-based coding, JBIG-2, Bit-plane encoding, Bit-allocation, Zonal Coding, Threshold Coding, JPEG, Lossless predictive coding, Lossy predictive coding and Motion Compensation.
- Wavelet based Image Compression-Wavelet series expansion, Discrete and Continuous Wavelet Transform, Fast Wavelet Transform, 2-D wavelet Transform, JPEG-2000 encoding and Digital Image Watermarking.
- Morphological Image Processing-Boundary Detection, Hole filling, Connected components, convex hull, thinning, thickening, skeletons, pruning and Reconstruction by dilation and erosion.
- Image Segmentation-Boundary detection-based techniques, Point, line detection, Edge detection, Hough transform, Thresholding, Otsu's method, Region-based segmentation and Watershed algorithm.

Course Specific Outcomes – B.Sc. Computer Science (General)

SEMESTER-I

CMSGCOR01T: Problem Solving with Computer

The course enables students to attain:

- An understanding of Computer Fundamentals and Basic Computer Organization - CPU, ALU, memory hierarchy, registers, I/O devices.
- An ability of problem solving, Problem definition, Program design, Debugging and Documentation.
- An ability to learn Techniques of Problem Solving by Flowcharting, decision table, algorithms, Structured programming concepts
- An understanding of Programming methodologies viz. top-down and bottom-up programming.

- An understanding of Python Interpreter, Python shell, Atoms, Identifiers and keywords, Literals, Strings and Operators.
- An ability to create Python Programs using Input and Output Statements, Control statements and Conditional Statement
- An understanding of Strings, Lists, Tuples, Dictionary, Date & Time, Modules, Defining Functions, Exit function, default arguments.
- An ability to develop Advanced Python Programs that includes Event Driven Programming and GUI Programming.

SEMESTER-II

CMSGCOR02T: Database Management Systems

The course enables students to attain:

- The fundamental knowledge of Database Management Systems and its architecture
- An understanding of Entity Relationship and Enhanced ER Modeling
- The knowledge of Schema Definition, constraints and object modeling.
- An understanding of relational algebra and SQLqueries.
- An understanding of functional dependencies and Normalization.

SEMESTER-III

CMSGCOR03T: Operating Systems

The course enables students to attain:

- The fundamental knowledge of System Software, Resource Abstraction and OS strategies
- An understanding of Multiprogramming, Batch, Time Sharing, Single user, Multiuser, Process Control & Real Time Systems.
- An understanding of basic OS functions, process modes, system calls and system programs.
- The knowledge of non-pre-emptive and pre-emptive process Scheduling strategies
- An understanding of memory allocation strategies-fixed partition, variable partition, paging and virtual memory
- The fundamental knowledge of shell and its types
- An ability to create and execute shell script
- An understanding of Shell variable, System calls, Pipes and Filters.
- An understanding of Decision making and functions in Shell Scripts
- The knowledge of Utility programs and Pattern matching utility

SEMESTER-IV

CMSGCOR04T: Computer System Architecture

The course enables students to attain:

- The fundamental knowledge of Logic gates, Boolean algebra, combinational circuits and sequential circuits.
- The knowledge of Data Representation and basic Computer Arithmetic
- A basic understanding of Computer Organization and Design (bus system, instruction set, timing and control, instruction cycle, memory reference, input-output and interrupt)
- An understanding of Central Processing Unit
- The knowledge of Instruction formats, addressing modes, instruction codes, machine language, assembly language and input output programming.
- The knowledge of Input-output Organization

SEMESTER-V

CMSGDSE02T: Discrete Structures

The course enables students to attain:

- The fundamental knowledge of Set, Relation and Function.
- An understanding of Pigeonhole Principle, Permutation and Combinations, Mathematical Induction, Principle of Inclusion and Exclusion.
- An understanding of Asymptotic Notations
- An ability to evaluate Recurrence Relations
- An understanding of Inference Theory
- A basic knowledge of Graphs Terminology and its representation
- An understanding of Graph Isomorphism, Graph Connectivity, Euler and Hamiltonian Paths and Circuits, Planar Graphs and Graph Coloring
- The fundamental knowledge of Tree and its properties, Spanning Trees.

SEMESTER-VI

CMSGDSE03T: Software Engineering

The course enables students to attain:

- The fundamental knowledge of Software Process (water fall, incremental, spiral, evolutionary, prototyping, object oriented)
- An understanding of Software requirement Analysis and feasibility studies
- The knowledge of Design Concepts and Principles
- An understanding of Software Configuration Management process, Version control, Change control Configuration audit and SCM standards.
- An ability to analyze S/W complexity, size measure, data and logic structure measure and information flow measure
- An ability to estimate Software Projects by implementing Empirical Estimation Models
- An understanding of software testing (black box testing, structural testing, regression testing, unit testing, integration testing, validation testing, system testing and debugging.
- An understanding of Reverse Engineering and Re-engineering