Semester 2 (1st year 2nd Semester)

Course Code	Course Title	Credit	Theory	Lab
CSE 1201	Data Structure	1	1	0

Outline: Introduction - Data Structures and Complexity of Algorithms, Time Space Tradeoff, Searching Techniques: Linear and Binary Searching; Sorting and Recursion - Discussion of Common Sorting Techniques: Insertion Sort, Selection Sort, Bubble Sort, Quick Sort, Merge Sort, Radix Sort; Factorial and Tower of Hanoi Problem; Linked Lists - Abstract Data Types, List ADTs, and Linked Lists: Singly, Two Way and Circular Linked Lists; Stacks and Queues - Stacks and Queues and their Implementation Strategies; Prefix, Infix and Postfix Expressions, their Transformation and Evaluation Algorithms; Hashing - Hash Indices and Hash Functions, Static and Dynamic Hashing, Collisions in Hash Indices and Collision Resolving Techniques; Trees - Tree Concepts, Binary Tree, BST, Heaps, Heap Sort, Huffman Encoding Technique, AVL Tree, B Tree and B+ Tree; Graphs - Graph Terminologies, Representing Graphs, Graph Searching: BFS and DFS, Shortest Path Problems, Minimum Spanning Tree, Minimum Spanning Tree Algorithms, and Topological Sorting; Problem Solving Strategy - Greedy Algorithms, Divide and Conquer Strategy, Dynamic Programming and Backtracking.

References:

1. *Data Structures*. Schaum's Outline Series.
2. E. Horowitz and S. Sahni, *Fundamentals of Data Structures*, London Pitman.
3. Robert L. Kruse, *Data Structures and Program Design*, Prentice Hall, 2nd Ed.

Course Code	Course Title	Credit	Theory	Lab
CSE 1202	Data Structure Lab	2	0	2

Outline: Introduction - Data Structures and Complexity of Algorithms, Time Space Tradeoff, Searching Techniques: Linear and Binary Searching; Sorting and Recursion - Discussion of Common Sorting Techniques: Insertion Sort, Selection Sort, Bubble Sort, Quick Sort, Merge Sort, Radix Sort; Factorial and Tower of Hanoi Problem; Linked Lists - Abstract Data Types, List ADTs, and Linked Lists: Singly, Two Way and Circular Linked Lists; Stacks and Queues - Stacks and Queues and their Implementation Strategies; Prefix, Infix and Postfix Expressions, their Transformation and Evaluation Algorithms; Hashing - Hash Indices and Hash Functions, Static and Dynamic Hashing, Collisions in Hash Indices and Collision Resolving Techniques; Trees - Tree Concepts, Binary Tree, BST, Heaps, Heap Sort, Huffman Encoding Technique, AVL Tree, B Tree and B+ Tree; Graphs - Graph Terminologies, Representing Graphs, Graph Searching: BFS and DFS, Shortest Path Problems, Minimum Spanning Tree, Minimum Spanning Tree Algorithms, and Topological Sorting; Problem Solving Strategy - Greedy Algorithms, Divide and Conquer Strategy, Dynamic Programming and Backtracking.

References:

1. *Data Structures*. Schaum's Outline Series.
2. E. Horowitz and S. Sahni, *Fundamentals of Data Structures*, London Pitman.
3. Robert L. Kruse, *Data Structures and Program Design*, Prentice Hall, 2nd Ed.

Course Code	Course Title	Credit	Theory	Lab
CSE 1203	Computer Organization	2	2	0

Outline: Introduction: Function and structure of a computer, Functional components of a computer, Interconnection of components, Performance of a computer; **Number representation and arithmetic**:
Binary, octal, and hexadecimal numbers, One's and two's complements and other representations,

Addition and subtraction; **Digital logic and integrated circuits**: Boolean algebra and truth tables, Boolean functions (Gates, Functions, Simplification), Integrated circuits (Combinational circuits - adders, shifters, decoders, multiplexers and ROM's; Flip-flops; Sequential circuits - registers, counters and RAM); **Representation of Instructions:** Machine instructions, Operands, Addressing modes, Instruction formats, Instruction sets, Instruction set architectures - CISC and RISC architectures; **Introduction to Assembly Language:** Programming with Assembly language, The assembly process, Linking and loading, Register-level debugging, **Processing Unit:** Organization of a processor - Registers, ALU and Control unit, Data path in a CPU, Instruction cycle, Organization of a control unit - Operations of a control unit, Hardwired control unit, Micro-programmed control unit; **Memory Subsystem:** Semiconductor memories, Memory cells - SRAM and DRAM cells, Internal Organization of a memory chip, Organization of a memory unit, Error correction memories, Interleaved memories, Cache memory unit - Concept of cache memory, Mapping methods, Organization of a cache memory unit, Fetch and write mechanisms, Memory management unit - Concept of virtual memory, Address translation, Hardware support for memory management; **Input/Output Subsystem:** Access of I/O devices, I/O ports, I/O control mechanisms - Program controlled I/O, Interrupt controlled I/O, and DMA controlled I/O, I/O interfaces - Serial port, Parallel port, PCI bus, SCSI bus, USB bus, Firewall and InfiniBand, I/O peripherals - Input devices, Output devices, Secondary storage devices; **Multiprocessing Systems:** Shared memory multiprocessor, Message-passing multiprocessor, Hardware multithreading

References:

1. D. A. Patterson and J. L. Hennessy, "Computer Organization and Design - The Hardware/Software Interface", Morgan Kaufmann, 1998.
2. C. Hamacher, Z. Vranesic and S. Zaky, "Computer Organization", McGraw-Hill, 2002.
3. W. Stallings, "Computer Organization and Architecture - Designing for Performance", Prentice Hall of India, 2002.
4. J .P. Hayes, "Computer Architecture and Organization", McGraw-Hill, 1998.

Course Code	Course Title	Credit	Theory	Lab
CSE 1204	Computer Organization Lab	1	0	1

Outline: Introduction: Function and structure of a computer, Functional components of a computer, Interconnection of components, Performance of a computer; **Number representation and arithmetic**:
Binary, octal, and hexadecimal numbers, One's and two's complements and other representations,

Addition and subtraction; **Digital logic and integrated circuits**: Boolean algebra and truth tables, Boolean functions (Gates, Functions, Simplification), Integrated circuits (Combinational circuits - adders, shifters, decoders, multiplexers and ROM's; Flip-flops; Sequential circuits - registers, counters and RAM); **Representation of Instructions:** Machine instructions, Operands, Addressing modes, Instruction formats, Instruction sets, Instruction set architectures - CISC and RISC architectures; **Introduction to Assembly Language:** Programming with Assembly language, The assembly process, Linking and loading, Register-level debugging, **Processing Unit:** Organization of a processor - Registers, ALU and Control unit, Data path in a CPU, Instruction cycle, Organization of a control unit - Operations of a control unit, Hardwired control unit, Micro-programmed control unit; **Memory Subsystem:** Semiconductor memories, Memory cells - SRAM and DRAM cells, Internal Organization of a memory chip, Organization of a memory unit, Error correction memories, Interleaved memories, Cache memory unit - Concept of cache memory, Mapping methods, Organization of a cache memory unit, Fetch and write mechanisms, Memory management unit - Concept of virtual memory, Address translation, Hardware support for memory management; **Input/Output Subsystem:** Access of I/O devices, I/O ports, I/O control mechanisms - Program controlled I/O, Interrupt controlled I/O, and DMA controlled I/O, I/O interfaces - Serial port, Parallel port, PCI bus, SCSI bus, USB bus, Firewall and InfiniBand, I/O peripherals - Input devices, Output devices, Secondary storage devices; **Multiprocessing Systems:** Shared memory multiprocessor, Message-passing multiprocessor, Hardware multithreading

References:

1. D. A. Patterson and J. L. Hennessy, "Computer Organization and Design - The Hardware/Software Interface", Morgan Kaufmann, 1998.
2. C. Hamacher, Z. Vranesic and S. Zaky, "Computer Organization", McGraw-Hill, 2002.
3. W. Stallings, "Computer Organization and Architecture - Designing for Performance", Prentice Hall of India, 2002.
4. J .P. Hayes, "Computer Architecture and Organization", McGraw-Hill, 1998.

Course Code	Course Title	Credit	Theory	Lab
STAT 1205	Probability and Statistics for Engineers-II	3	3	0

Course Outline: Hypothesis Testing: Tests Concerning the Mean of a Normal Population, Testing the Equality of Means of Two Normal Populations, Hypothesis Tests Concerning the Variance of a Normal Population, Hypothesis Tests in Bernoulli Populations and Tests Concerning the Mean of a Poisson Distribution. Regression and Correlation Analysis: Least Squares Estimators of the Regression Parameters, Distribution of the Estimators, Statistical Inference about the Regression Parameters, Coefficient of Determination and Sample Correlation Coefficient, Analysis of Residuals, Transforming to Linearity, Weighted Least Squares, Polynomial Regression, Multiple Linear Regression, Logistic Regression Models for Binary Output Data and Correlation Analysis. Analysis of Variance: One-way Analysis of Variance, Two-Factor Analysis of Variance: Introduction and Parameter Estimation, Testing Hypotheses and Two-way Analysis of Variance with Interaction Problems. Goodness of Fit Tests and Categorical Data Analysis: Goodness of Fit Tests when All Parameters are Specified, Goodness of Fit Tests when All Parameters are Unspecified, Tests of Independence in Contingency Tables, Tests of Independence in Contingency Tables Having Fixed Marginal Totals and Kolmogorov-Smirnov Goodness of Fit Test for Continuous Data. Nonparametric Hypothesis Tests: Sign Test, Signed Ranked Test, Two-Sample Problem and Runs Tests for Randomness. Quality Control: Control Charts for Average Values, The X-Control Chart, S-Control Charts, and Control Charts for the Fraction Defective, Control Charts for Number of Defects and Other Control Charts for Detecting Changes in the Population Mean.

References:

1. Sheldon M. Ross, Introduction to Probability and Statistics for Engineers and Scientists, Elsevier/Academic Press, 3rd Ed.
2. Douglas C. Montgomery and George C. Runger, Applied Statistics and Probability for Engineers, John Wiley and Son, 4th Ed.
3. Murray R Spiegel, John J Schiller, R Alu Srinivasan, Schaum's Outline: Probability and Statistics, McGraw Hill, 3rd Ed.

Course Code	Course Title	Credit	Theory	Lab
MATH 1207	Ordinary Differential Equations	3	3	0

Course Outline: Differential Equations and Mathematical Modeling, Initial Value Problem, Separable Differential Equations, Exact Differential Equations, Linear Differential Equations, Bernoulli Equation, Homogeneous Linear Equations of Second Order, Second Order Homogeneous Equations with Constant Coefficients, Euler-Cauchy Equation, Existence and Uniqueness Theory, Non-homogeneous Equations, Solution by Undetermined Coefficients, Solution by Variation of Parameters, Higher-Order Linear Differential Equations, Higher-Order Homogeneous Equations with Constant Coefficients, and Higher-Order Non-homogeneous Equ

ations, Vectors, Matrices, and Eigenvalues, Homogeneous Systems with Constant Coefficients, Critical Points, Criteria for Critical Points, Stability, Qualitative Methods for Nonlinear Systems, Non-homogenous Linear Systems, Laplace Transform, Inverse Transform, Transforms of Derivatives and Integrals, Differentiation and Integration of Transforms, Convolution, and Partial Fractions, System of Differential Equations.

References:

1. Erwin Kreyszig, *Advanced Engineering Mathematics*, John Wiley & Sons, 8th Ed.
2. S.L. Ross, *Differential Equations.*
3. Earl A. Coddington, *An Introduction to Ordinary Differential Equations*, Dover Publications, Unabridged Ed.
4. Morris Tenenbaum and Harry Pollard, *Ordinary Differential Equations*, Courier Dover Publications, 1985 Ed.

Course Code	Course Title	Credit	Theory	Lab
GE 1209	History of Emergence of Bangladesh	3	3	0

Course Outline:

Partition of Bengal in 1905, Devoid of Partition in 1911 and its Reactions.

Partition of the Sub-Continent 1947, Structure of Pakistan, Disparity, the Language Movement and the Rule of Ayub-Yahia Khan (1958-1971.

a. Lahore Resolution,1940
b. The creation of Pakistan 1947
c. Central and Provincial Structure
d. Economic, Social and Cultural Disparity
e. Misrule of Pakistan and Struggle for Democratic Politics
f. The Language Movement : Context and Phases
g. United Front of Haque-Vasani-Suhrawardi : Election of 1954 and its Consequences
h. Ayub Khan's Rise to Power and Characteristics of His Rule (Political Repression, Basic Democracy, Islamisation
i. Ayub Khan and Yahia Khan's Rule, Abolition of One Unit, Universal Suffrage, LFO

Rise of Nationalism and the Movement for Self- Determination.

a. The Six Point Movement of Sheikh Mujibur Rahman
b. Reactions, Importance and Significance of the Six Point Movement
c. The Agartala Case, 1968
d. Students' 11-Points Movement
e. The Mass-Upsurge of 1969

Election of 1970, Non-cooperation Movement of March 1971 and the Declaration of Independence by Bangabandhu.

a. Election Result and Central's Refusal to Comply
b. The Non-cooperation Movement, the 7th March Address, Operation Searchlight

c. Declaration of Independence by Bangabandhu and His Arrest
d. The Proclamation of Independence and the Formation of Bangladesh Government

References:

1.Harun-or-Rashid : *Foreshadowing of Bangladesh*
2.†ejvj †gvnv¤§` :¯^vaxb evsjv †eZvi †K›`ª
3.gI`y` Avn‡g` :†kL gywReyi ingv‡bi kvmbKvj
4.gybZvmxi gvgyb :¯^vaxb evsjv‡`‡ki Afz¨`‡qi BwZnvm
5.†gvt gvneyei ingvb :evsjv‡`‡ki BwZnvm, 1947-1971
6.†kL gywReyi ingvb : Amgvß AvZ¥Rxebx
7.ˆmq` AvwZKzj Bmjvg I Ab¨vb¨ :¯^vaxb evsjv‡`‡ki Afz¨`‡qi BwZnvm

Course Code	Course Title	Credit	Theory	Lab
SE 1213	Object Oriented Concepts I	2	2	0

Course Outline: Object Oriented Concepts - Introduction to Object Oriented Concepts – Procedural vs Object Oriented (OO) Programming, What is an Object – Object Data and Behavior, What is a Class

– Attributes, Methods and Messages, Using UML to model a Class Diagram, Encapsulation and Data Hiding: Interfaces and Implementations, Inheritance: Super classes and Subclasses, Abstraction and Is-a Relationships; Polymorphism, Composition: Abstraction and Has-a Relationships; How to think in terms of Objects – Interface vs Implementation, Abstract thinking when designing Interfaces and Giving the user minimal Interface possible; Object Oriented concepts in details – Constructors: Default constructor, When is a constructor called, Using multiple constructors and The design of constructors, Error handling and The concept of scope; The Anatomy of a Class – The Name, Comments, Attributes, Constructors, Accessors, Public Interface methods and Private implementation methods; Class Design Guidelines – Modeling Real World Systems, Identifying Public Interfaces, Designing Robust Constructors, Designing Error Handling to a Class, Documenting a Class and Using Comments, Designing with Reuse, Extensibility, Maintainability in Mind and Using Object Persistence; Designing with Objects – Proper Analysis, Statement of Work, Requirements Collection, Prototype of User Interface, Identifying the Classes, Determining the responsibilities of Each Class, Class Collaboration, Class Model to Describe the System; Mastering Inheritance – Reusing Objects, Generalization and Specialization, How Inheritance weakens Encapsulation; Frameworks and Reuse – When should we Reuse, Frameworks, Contract: Abstract Classes and Interfaces.

Programming lessons - Introduction to Java – Java Virtual Machine (JVM) and Java Runtime (JRE), Java Development Kit (JDK), Integrated Development Environment (IDE) for Java, Java installation, Hello World! Program, compiling and running Java program, using Java classpath and JVM Architecture; Java syntax – Package, Import, Class, Fields, Methods, Constructors, Primitive data types, Strings and literal, Wrapper class, Nonexistence type: null. Object Oriented Programming (OOP) - The students will implement each of the object oriented concepts which are discussed in the class. Java features to support practical OOP – String Operations: String creations and operations, immutability property of String, String comparison and searching, String buffers and builders; Java I/O: Streams, Input and Output Stream, File, Path, Directory

and tree; Exception handling: try and catch, checked exception vs unchecked exceptions, throw and throws, Common exception and User defined exceptions; Logger and Debugging: Logger, Log levels, Formatters and Filters, Logger Handlers and Manager, Configuration, Introduction to Debugging and Debugging Workflow.

References:

1. The Object Oriented Thought Process, Matt Weisfeld, Addison-Wesley
2. Java How to Program, Paul Deitel and Harvey Deitel, McGraw Hill
3. Java: The Complete Reference, Herbert Schildt, McGraw Hill

Course Code	Course Title	Credit	Theory	Lab
SE 1214	Object Oriented Concepts I Lab	1	0	1

Course Outline: Object Oriented Concepts - Introduction to Object Oriented Concepts – Procedural vs Object Oriented (OO) Programming, What is an Object – Object Data and Behavior, What is a Class

– Attributes, Methods and Messages, Using UML to model a Class Diagram, Encapsulation and Data Hiding: Interfaces and Implementations, Inheritance: Super classes and Subclasses, Abstraction and Is-a Relationships; Polymorphism, Composition: Abstraction and Has-a Relationships; How to think in terms of Objects – Interface vs Implementation, Abstract thinking when designing Interfaces and Giving the user minimal Interface possible; Object Oriented concepts in details – Constructors: Default constructor, When is a constructor called, Using multiple constructors and The design of constructors, Error handling and The concept of scope; The Anatomy of a Class – The Name, Comments, Attributes, Constructors, Accessors, Public Interface methods and Private implementation methods; Class Design Guidelines – Modeling Real World Systems, Identifying Public Interfaces, Designing Robust Constructors, Designing Error Handling to a Class, Documenting a Class and Using Comments, Designing with Reuse, Extensibility, Maintainability in Mind and Using Object Persistence; Designing with Objects – Proper Analysis, Statement of Work, Requirements Collection, Prototype of User Interface, Identifying the Classes, Determining the responsibilities of Each Class, Class Collaboration, Class Model to Describe the System; Mastering Inheritance – Reusing Objects, Generalization and Specialization, How Inheritance weakens Encapsulation; Frameworks and Reuse – When should we Reuse, Frameworks, Contract: Abstract Classes and Interfaces.

Programming lessons - Introduction to Java – Java Virtual Machine (JVM) and Java Runtime (JRE), Java Development Kit (JDK), Integrated Development Environment (IDE) for Java, Java installation, Hello World! Program, compiling and running Java program, using Java classpath and JVM Architecture; Java syntax – Package, Import, Class, Fields, Methods, Constructors, Primitive data types, Strings and literal, Wrapper class, Nonexistence type: null. Object Oriented Programming (OOP) - The students will implement each of the object oriented concepts which are discussed in the class. Java features to support practical OOP – String Operations: String creations and operations, immutability property of String, String comparison and searching, String buffers and builders; Java I/O: Streams, Input and Output Stream, File, Path, Directory and tree; Exception handling: try and catch, checked exception vs unchecked exceptions, throw and throws, Common exception and User defined exceptions; Logger and Debugging: Logger,

Log levels, Formatters and Filters, Logger Handlers and Manager, Configuration, Introduction to Debugging and Debugging Workflow.

References:

1. The Object Oriented Thought Process, Matt Weisfeld, Addison-Wesley
2. Java How to Program, Paul Deitel and Harvey Deitel, McGraw Hill
3. Java: The Complete Reference, Herbert Schildt, McGraw Hill