Institute of Information Technology (IIT)

Noakhali Science and Technology University

BACHELOR OF SCIENCE IN SOFTWARE ENGINEERING (BSSE)
January 2020

iit.nstu.edu.bd

Introduction

Institute of Information Technology (IIT) is one of the most glorious Institutes in Noakhali Science and Technology University (NSTU). On 20th August 2017, in the 36th Academic Council meeting of Noakhali Science and Technology University, the program of Bachelor of Science in Software Engineering (BSSE) has been introduced under the leadership of founder Director of Institute of Information Technology (IIT) **Mr. Mohammad Nuruzzaman Bhuiyan.** At present, **Mr. Md. Auhidur Rahman** is the acting director of IIT since May 2019.

BSSE program is a four-year industry-oriented program. Three years of study is followed by a half-year industry placement before the final semester. Students will be assisted by the Institute in their placement for the internship. They will integrate their industry experience within their studies upon return for their final semester. This will develop students in strong communication skills together with an outward worldly focus, positive personally and business attitudes. BSSE program will enhance student's capability and competence to deliver at different technical roles and management positions.

This program will cover a wide range of software engineering topics for a comprehensive coverage of modern software and techniques. For graduation, students have to complete total 150.0 credit hour. This program offers a common first year which consists of mathematics, sciences, computing, engineering principles, communications and design. BSSE program will provide students with a sloid foundation of engineering training and introducing in a variety of engineering disciplines. Students will also gain knowledge in areas such as computer programming, object-oriented methodology, software design, software validation and verification, software security and computer networks in the second and third year. In fourth year first semester, all the students will send in an industry for Industrial Training.

Program Description

Software engineering is the application of sound engineering principles and techniques to the analysis, design, development, testing and management of software systems. It is an interdisciplinary study of integrating traditional computer science which focuses on software development and related theoretical issues with engineering which emphasizes on designing and building complex, safe, reliable software for general use. Software Engineering is an Engineering

discipline whose focus is the cost-effective development of high quality software systems. It has real-life implications in many industries including medical, communications business, military, aerospace, scientific and general computing. Using principles and techniques of computer science, engineering and mathematical analysis, software engineers empower computers with innovative applications to perform tasks smarter, faster and better. Institute of information Technology (IIT) launces Bachelor of Science program in software Engineering form the academic year of 2017-18.

Throughout this program, we will provide students with a strong foundation in software engineering using a combination of classroom study, software development experience and design projects. Hence the program blends engineering principles, computing skills, project leadership and software construction to supply students with a comprehensive understanding of the field and to prepare graduates for the workforce or future study.

This program is designed around a set of core courses that introduces the fundamentals of software engineering, followed by a broader range of courses. Students could choose to augment their core with more Software Security-oriented courses (e.g. Software Security and Information Security), Data Science courses (e.g. Data Mining, Big Data and Large-scale Computing), Web Services and Applications oriented courses (e.g. Web programming, User Interface Design and Evaluation), or Graphics and Game related courses (e.g. animation for computer games, Artificial Intelligence for Games). Each of these areas is covered by a dedicated set of core and extended courses. In short, by providing a careful balance between theory and practice, the program will prepare students for central software positions in the software industry, government and institutions where software engineering has become a key activity.

Program Objectives

Information and Communication Technology (ICT) sector being one of the most knowledge-intensive branches of the economy, there is demand for human resource in the software engineering discipline in both public and private sectors. Software engineering is a field that deals with high-level designs and solutions that guide the development of specific software projects or products. The program has been tailor-made to produce graduates who are proficient in developing software according to industry standards in terms of methodologies and technologies. The course provides students with both theoretical knowledge and practical skills in areas such as software

development, integration & testing and software project management. Graduates will possess the engineering skills required to design and implement software systems. The scheme of study offers graduates essential technical and soft skills to seamlessly make the transition from University to the software development industry and adapt to a professional environment.

The program is in line with international recommendations of computing curricula for Undergraduate Degree Programs in Software Engineering and designed in collaboration with software industry. This includes:

a) Foundation: Graduates shall have strong foundation in science and mathematics. He can apply this fundamental knowledge in developing software engineering tasks.

b) Development: Graduates can effectively apply software engineering practice over the entire system life cycle. This includes requirements engineering, analysis, prototyping, design, risks involved in software and embedded systems.

c) Process: Graduates know classical and evolving software engineering methods, can select and tailor appropriate methods for projects and can apply them as both team members and managers to achieve project goals.

d) Professionalism: Graduates are knowledgeable of the ethics, professionalism and cultural diversity in the work environment.

e) Quality: Graduates can apply basic software quality assurance practices to ensure that software designs, development and maintenance meet or exceed applicable standards.

f) Presentation: Graduates have effective written and oral communication skills. Graduates can prepare and publish the necessary documents required throughout the project life cycle. Graduates can effectively contribute to project discussions, presentations and reviews.

g) Growth: Graduates understand the need for lifelong learning and can readily adapt to new software engineering environments.

Program Mission

The primary mission of Bachelor of Science in Software Engineering program is to develop professionals who can define, design, and develop high-quality software systems within resource constraints. We accomplish this mission through our undergraduate and graduate programs in software engineering and the research activities of our faculties working along with their student's research team.

Program Vision

Graduates of Bachelor of Science in Software Engineering program will be recognized as innovative leaders in the field of computer science and software engineering by their work in software development in a myriad of application areas and though their work in advanced study and research. The faculties will continue to be known for their passion for teaching these students and for their knowledge for their passion for teaching these students and for their knowledge, expertise and innovation in advancing the frontiers of knowledge in computer science and software frontiers of knowledge in computer science and software engineering.

Faculties

Bachelor of Science in Software Engineering Program was launched under the Institute of Information Technology (IIT). The Institute consists of four renowned faculties. They are listed below:

S/N	Name	Designation	Research Area
1.	Md. Auhidur Rahman	Assistant Professor & Director (Acting)	IoT, Fog, Cloud Computing
2.	Mohammad Nuruzzaman Bhuiyan	Assistant Professor (on Study Leave)	IoT, Cyber Security, E-commerce
3.	Dipanita Saha	Assistant Professor	Bio Informatices, IoT
4.	Falguni Roy	Assistant Professor	Information Retrieval, Data Mining
5.	Dipok Chandra Das	Lecturer	Data Science, Software Analytics
6.	Md. Iftekhar Alam Efat	Lecturer	Software Reusability, Software Architecture, Machine Learning, Big Data & Data Science, IoT
7.	Tasniya Ahmed	Lecturer (on Maternity Leave)	N/A

Students

In both 2017-2018 and 2018-2019 Academic sessions, about 60 (sixty) students were admitted into this program according to merit position, comprising about 30 (thirty) students in each session.

Degree Requirements

In order to qualify for the BSSE degree, a student has to meet the following requirements:

- Completion of minimum 144 credits including an internship program.
- A minimum of grade C+ in a Comprehensive Examination.
- Passing of all courses individually with at least D grade.
- Grade Point Average (GPA) of 2.5 or above.

Course Load

BSSE is full-time course of study and each student must take 18 credits in each semester. Any student failing to take 18 credits in a regular semester will stand withdrawn from the program for that particular semester. An exception to this rule may be made only by the academic committee of IIT. The full-time course load may be relaxed for transfer students or who are enrolling again after withdrawal.

Break down of a Semester

The BSSE program is a four-year program consisting of eight semesters. Each semester has duration of six months; the break down is presented as follows:

Weeks	Purpose
14 weeks	Scheduled classes
1 Week	Preparation time for examinations
3 weeks	Semester final examination
3 weeks	Result publication

4 weeks	Vacation and holidays
1 week	Supplementary examinations

* Less than 75% attendance will be treated as non-collegiate student. A non-collegiate student has to apply to the Director, Regular Program Office to sit for the examination and upon the approval of academic committee, IIT; he/she has to deposit TK. 3000/- (Three Thousand) as fine as per university rules.
* Below 60% attendance, a student should not be allowed to sit for the examination as per university rules.

Unfair Means

Students are strictly forbidden from adopting unfair means. Students who will adopt unfair means will be punished as per rules of Noakhali Science and Technology University.

Grading System & Definition of a Credit

The credit is defined as follows:

Class	Class Type	Hours/Week
1	Credit Theory	1hr
1	Credit Laboratory	2hr

Most of the courses will consist of both theoretical classes and laboratory works.

* The total number of credits of a course will be distributed for both theoretical and laboratory works.

Letter Grade, Grade Points and their Meaning

Grades in each course will be assigned (in according with the rules NSTU/UGC) as mentioned below.

Marks	Letter Grade	Numeric Grade	Comments
80% or above	A+	4.00	Excellent
>=75% but < 80%	A	3.75	Better
>=70% but < 75%	A-	3.50	Good
>=65% but <70%	B+	3.25	Above average
>=60% but <65%	B	3.00	Average
>=55% but 60%	B-	2.75	Below average
>=50% but <55%	C+	2.50	Satisfactory
>=45% but <50%	C	2.25	Not satisfactory
>=40% but <45%	D	2.00	Pass
Less than 40%	F	0.00	Fail
	I		Incomplete
	W		Withdrawn

Promotion to the next Semester

* The overall CGPA obtained by a student in the previous semester must not be less than 2.5
* A student will have to secure at least grade D in each course in the previous semester.
* Students who achieved overall CGPA 2.5 but F in any course will have to sit for supplementary exam and he/she will get no more than B+ in that course.
* Students failed to get promoted will retake that semester with the following batch.
* However, a student may retake only those courses for which he/she got ‘F’ grade.

Comprehensive Examination

A Comprehensive Examination is taken to evaluate to student’s understanding of their major areas of study (Software Engineering courses). Students must earn a minimum grade ‘C+’ in this examination. The Comprehensive Examination grade is shown on the Grade Sheet but is not included in the calculation of CGPA.

The Comprehensive Examination is usually taken two weeks before the end of the final semester. If a student fails the Comprehensive Examination, he/she may sit for a retake which is allowed only once, unless otherwise decided by the Academic Committee.

Internship Program

To gain the industry experience, students of seventh semester will be assigned with a well-known organization of Information Technology industry. Students, upon completion of 108 credits (36 different courses), will have an acceptable theoretical knowledge. With such background, students will move to industry to implement the knowledge he or she gathered and at the same time to be aware of the industry trend and there working environment. After completion of the internship, students will be back to the institution and have to present a report on their domain of work in the respective organization. Students will also be evaluated form the assigned organization.

Applicability of the Curriculum and Rules

The Institute of Information Technology reserves the right to make, at any time without notice, changes to program, courses, statements contained in this booklet. No responsibility will be borne by Noakhali Science and Technology University or by the Institute of Information Technology (IIT) for any adjustment or expenses resulting from such changes.

Final Remarks

The software industry is always looking for qualified engineers from the Universities. Instead of going with traditional systems, IIT is proposing an industry-oriented program. IIT believes BSSE will work more like a bridge between the software industry and educational institutes.

The courses are designed in such a way that within the first six semesters each student will acquire the knowledge to go to the industry and work. Upon completion of 6 months of internship experiences with the theory. At the end, a BSSE fresh graduate will be a complete software engineer who can go start working in the industry.

Software Engineering is an important stream of typical computer science, what IIT is planning to address. There are other streams such as database, networking etc. which have acute market demands. The future goal of IIT is set to produce quality practitioners in those stream as well.

Institute of Information Technology (IIT)

Noakhali Science and Technology University

Bachelor of Science in Software Engineering (BSSE)

Year 1 Term 1

Course Code	Course Title	Credit	Theory	Lab
CSE 1101	Structured Programming	1	1	0
CSE 1102	Structured Programming Lab	2	0	2
CSE 1103	Discrete Mathematics	3	3	0
STAT 1105	Probability and Statistics for Engineers-I	3	3	0
MATH 1107	Calculus and Analytical Geometry	3	3	0
GE 1109	Soft Skill Communication	3	3	0
GE 1111	Technology and Society	2	2	0
GE 1112	Technology and Society Lab	1	0	1
SE 1113	Introduction to Software Engineering	3	3	0
9 Courses		**21**	**18**	**3**

Year 1 Term 2

Course Code	Course Title	Credit	Theory	Lab
CSE 1201	Data Structure	1	1	0
CSE 1202	Data Structure Lab	2	0	2
CSE 1203	Computer Organization	2	2	0
CSE 1204	Computer Organization Lab	1	0	1
STAT 1205	Probability and Statistics for Engineers-II	3	3	0
MATH 1207	Ordinary Differential Equations	3	3	0
GE 1209	History of Emergence of Bangladesh	3	3	0
GE 1211	Bengali Literature	3	3	0
SE 1213	Object Oriented Concepts I	2	2	0
SE 1214	Object Oriented Concepts I Lab	1	0	1

10 Courses		**21**	**17**	**4**

Year 2 Term 1

Course Code	Course Title	Credit	Theory	Lab
CSE 2101	Algorithm Analysis	2	2	0
CSE 2102	Algorithm Analysis Lab	1	0	1
SE 2103	Theory of Computation	2	2	0
SE 2104	Theory of Computation Lab	1	0	1
CSE 2105	Computer Networks	2	2	0
CSE 2106	Computer Networks Lab	1	0	1
MATH 2107	Numerical Analysis for Engineers	2	2	0
MATH 2108	Numerical Analysis for Engineers Lab	1	0	1
SE 2109	Object Oriented Concepts II	2	2	0
SE 2110	Object Oriented Concepts II Lab	1	0	1
SE 2112	Software project Lab I	3	0	3
11 Courses		**18**	**10**	**8**

Year 2 Term 2

Course Code	Course Title	Credit	Theory	Lab
CSE 2201	Operating Systems and System Programming	2	2	0
CSE 2202	Operating Systems and System Programming Lab	1	0	1
GE 2203	Business Psychology	3	3	0
CSE 2205	Information Security	2	2	0
CSE 2206	Information Security Lab	1	0	1
CSE 2207	Database Management System-I	2	2	0
CSE 2208	Database Management System-I Lab	1	0	1
SE 2209	Software Requirements Spec. and Analysis	2	2	0
SE 2210	Software Requirements Spec. and Analysis Lab	1	0	1
BUS 2211	Business Studies for Engineers	3	3	0
10 Courses		**18**	**14**	**4**

Year 3 Term 1

Course Code	Course Title	Credit	Theory	Lab
SE 3101	Professional Ethics for Information Systems	3	3	0
CSE 3103	Web Technology	1	1	0
CSE 3104	Web Technology Lab	2	0	2
CSE 3105	Data Science and Analytics – DBMS II	1	1	0
CSE 3106	Data Science and Analytics – DBMS II Lab	2	0	2
BUS 3107	Business Communications	2	2	0
BUS 3108	Business Communications Lab	1	0	1
SE 3109	Design Pattern	2	2	0
SE 3110	Design Pattern Lab	1	0	1
SE 3112	Software Project Lab II	3	0	3
10 Courses		**18**	**9**	**9**

Year 3 Term 2

Course Code	Course Title	Credit	Theory	Lab
CSE 3201	Distributed Systems	1	1	0
CSE 3202	Distributed Systems Lab	2	0	2
SE 3203	Software Metrics	2	2	0
SE 3204	Software Metrics Lab	1	0	1
SE 3205	Software Security	2	2	0
SE 3206	Software Security Lab	1	0	1
CSE 3207	Artificial Intelligence	2	2	0
CSE 3208	Artificial Intelligence Lab	1	0	1
SE 3209	Software Testing and Quality Assurance	2	2	0
SE 3210	Software Testing and Quality Assurance Lab	1	0	1
SE 3211	Software Design and Architecture	2	2	0
SE 3212	Software Design and Architecture Lab	1	0	1
12 Courses		**18**	**11**	**7**

Year 4 Term 1

Course Code	Course Title	Credit	Theory	Lab
SE 4100	Internship	18	0	18
1 Courses		**18**	**0**	**18**

Year 4 Term 2

Course Code	Course Title	Credit	Theory	Lab
SE 4202	Project	6	0	6
SE 4203	Software Maintenance	2	2	0
SE 4204	Software Maintenance Lab	1	0	1
SE 4205	Software Project Management	2	2	0
SE 4206	Software Project Management Lab	1	0	1
SE/CSE 42XX	Elective	3	2	1
SE/CSE 42XX	Elective	3	2	1
5 Courses		**18**	**8**	**10**

Summary

Title	No of Subject	Credit	
CSE	14	42	
MATH	3	9	
STAT	2	6	
GE	5	15	
SE	17+1+1	51+18+6 = 75	
BUS	1	3	

Year 1 Term 1

CSE 1101	Structured Programming	1	1	0

Course Code	Course Title	Credit	Theory	Lab
CSE 1101	Structured Programming	1	1	0

Course Outline: Fundamentals of C programming; Introducing C's Program Control Statements; Data types, Variables and Expressions; Exploring Arrays and Strings; Understanding Pointers and Functions; Console and File I/O; Structures and Unions.

References:

1. Teach Yourself C, Herbert Schildt, McGraw Hill
2. C: The Complete Reference, Herbert Schildt, McGraw Hill
3. Schaum's Outline of programming with C, McGraw Hill

Course Code	Course Title	Credit	Theory	Lab
CSE 1102	Structured Programming Lab	2	0	2

Course Outline: Fundamentals of C programming; Introducing C's Program Control Statements; Data types, Variables and Expressions; Exploring Arrays and Strings; Understanding Pointers and Functions; Console and File I/O; Structures and Unions.

References:

1. Teach Yourself C, Herbert Schildt, McGraw Hill
2. C: The Complete Reference, Herbert Schildt, McGraw Hill
3. Schaum's Outline of programming with C, McGraw Hill

CSE 1103	Discrete Mathematics	3	3	0

Course Code	Course Title	Credit	Theory	Lab
CSE 1103	Discrete Mathematics	3	3	0

Course Outline: The Foundations: Logic and Proofs: propositional logic, applications of propositional logic, propositional equivalences, predicates and quantifiers, nested quantifiers, rules of inference, introduction to proofs; **Basic Structures:** Sets, Functions, Sequences, Sums, and Matrices; **Number Theory:** The division algorithm, divisibility and the euclidean algorithm, prime numbers,

congruence, applications of congruence; **Induction and Recursion:** Mathematical Induction, Recursive Definitions and Structural Induction, Program Correctness; **Counting:** The addition and multiplication rules, The principle of Inclusion-Exclusion, The pigeon-hole principle, permutations, combinations, Generalized Permutations and Combinations, Generating Permutations and Combinations; **Relations and Functions:** Symmetry, transitivity, reflexivity, equivalence classes, congruence, closure of relations, partial orderings; **Graphs:** Graphs and Graph Models, Graph Terminology and Special Types of Graphs, Representing Graphs and Graph Isomorphism, Connectivity, Euler and Hamilton Paths; **Trees:** Introduction to Trees, Tree Traversal, Spanning Trees.

References:

1. Discrete Mathematics and its Applications, Seventh Edition by Kenneth H. Rosen.

Course Code	Course Title	Credit	Theory	Lab
STAT 1105	Probability and Statistics for Engineers-I	3	3	0

Course Outline: Introduction to Statistics: Concept of Data and Variables, Data Collection and Descriptive Statistics, Inferential Statistics, Populations and Samples; **Descriptive Statistics:** Frequency Tables and Graphs, Relative Frequency Tables and Graphs, Grouped Data, Histograms, Ogives, Stem and Leaf Plots, Sample Mean, Sample Median, Sample Mode, Sample Variance and Standard Deviation, Sample Percentiles and Box Plots, Chebyshev's Inequality, Normal Data Sets,

Paired Data Set and Sample Correlation Coefficient; **Elements of Probability:** Basic Terminology in Probability, Sample Space and Events, Venn Diagrams and Algebra of Events, Axioms of Probability,

Conditional Probability, Bayes' Theorem and Independent Events; **Random Variables and Expectation:** Random Variables, Types of Random Variables, Jointly Distributed Random Variables, Expectation, Property of Expected Values, Use of Expected Values in Decision Making, Variance, Covariance and Variance of Sums of Random Variables and Moment Generating Functions; **Special Random Variables:** Binomial Random Variables, Poisson Random Variables, Uniform Random Variables, Normal Random Variables, Exponential Variables, Gamma Distribution, Chi-Square Distribution, t-Distribution and F-Distribution; **Distributions of Sampling Statistics:** Central Limit Theorem, Sampling Distribution for Normal Population, and Sampling from a Finite Population; **Parameter Estimation:** Maximum Likelihood Estimators, Interval Estimates, Estimating the difference in Means of Two Normal Population, Approximate Confidence Interval for the Mean,
Confidence Interval of the Mean of the Exponential Distribution and Bayes' Estimator.

References:

1. Sheldon M. Ross, Introduction to Probability and Statistics for Engineers and Scientists, Elsevier/Academic Press, 3rd Ed.
2. M. Nurul Islam, An Introduction to Statistics and Probability, Book World, 3rd Edition.
3. Lipschutz, Lipschutz Seymour, 2000 Solved Problems in Discrete Mathematics, McGraw-Hill, 1st Ed

Course Code	Course Title	Credit	Theory	Lab
MATH 1107	Calculus and Analytical Geometry	3	3	0

Course Outline: Basic Concepts: Real Numbers and Real Lines, Polar Coordinates, Parametric Equations, Functions, Algebra of Functions, Inverse Functions, Quadratic Functions, Shifting Graphs, Trigonometric Functions, Complex Numbers, Inequalities, Infinite Series and Sequences, Taylor Series, Rate of Change and Limit, Rules of Finding Limits, Formal Definition of Limit, Extension of the Limit Concepts, L'Hospitals Rule, Continuity, Tangent Lines; **Differential Calculus:** The Derivatives of a Function, Differentiation Rules, Rates of Change, Derivatives of Trigonometric Functions, Chain Rule Differentiation, Implicit Differentiation and Rational Exponents, Related Rates of Change, Extreme Values of Functions, Mean Value Theorem, First Derivative and Second Derivative Tests for Extreme Values, Optimization, Linearization and Differentials and Newton's Method; **Integral Calculus:** Indefinite Integrals, Integration by Substitution, Riemann Sums, Definite Integral, Fundamental Theorem of Calculus, Mean Value Theorem, Substitution in Definite Integrals, Areas between Curves, Finding Volumes by Slicing, Volumes of Solids of Revolution, Cylindrical Shells, Lengths of Plane Curves, Areas of Surfaces of Revolution, Moments and Center of Mass, Fluid Pressures and Forces, Integration by Parts, Improper Integrals, Multiple Integrals and Line Integrals; **Linear Algebra and Vector Calculus:** Matrices, Operation on Matrices, Inverse of a Matrix, Rank of Matrix, Determinant, Vectors, and Solutions of System of Linear Equations, and Eigen value Problems.

References:

1. G.B. Thomas and R.L. Finney, *Calculus and Analytical Geometry*, Addison Wesley, 9th Ed.
2. Erwin Kreyszig, *Advanced Engineering Mathematics*, John Wiley & Sons, 9th Ed.

GE 1109	Soft Skill Communication	3	3	0

Course Code	Course Title	Credit	Theory	Lab
GE 1109	Soft Skill Communication	3	3	0

Course Outline: The elements of Communication: The importance of communication through English at the present time, The process of communication and factors that influence communication sender, receiver, channel, code, topic, message, context, feedback, noise, filters & barriers, The importance of audience and purpose, The information gap principle: given and new information, information overload, Verbal and non-verbal communication: body language, Comparing general communication and business communication, the sounds of English, Review of English grammar**.**

References:

1. An introduction to Professional English and Soft Skills by B. K. Das et al., Cambridge University Press .
2. Technical Communication: Principles and Practice, Second Edition by Meenakshi Raman and Sangeeta Sharma, Oxford Publications.
3. Effective Technical Communication by M Ashraf Rizvi, The McGraw-Hill companies.
4. Understanding Body Language by Alan Pease.
5. Communicative Grammar of English by Geoffrey Leech and Ian Svartik.

6. Better English Pronunciation by J.D.O'Connor.
7. English Grammar by S.PitCorder
8. English Grammar by Wren and Martin. This is not the end of the list other books may also be referred.

GE 1111	Technology and Society	2	2	0

Course Code	Course Title	Credit	Theory	Lab
GE 1111	Technology and Society	2	2	0

Course Outline: Technology and Society: Nature of technology, social forces that affect its adoption; impact on society, innovation, within historical and contemporary contexts, the societal implications of technology, Effects of technological factors on social life and Influence of Technology on Social Institution, public policy implications of innovation, changing nature of technology and its impact on society.

References:

1. Society and Technological Change. 6th edition. Worth Publishers Inc.

Course Code	Course Title	Credit	Theory	Lab
GE 1112	Technology and Society Lab	1	0	1

Course Outline: Technology and Society: Nature of technology, social forces that affect its adoption; impact on society, innovation, within historical and contemporary contexts, the societal implications of technology, Effects of technological factors on social life and Influence of Technology on Social Institution, public policy implications of innovation, changing nature of technology and its impact on society.

References:

1. Society and Technological Change. 6th edition. Worth Publishers Inc.

SE 1113	Introduction to Software Engineering	3	3	0

Course Code	Course Title	Credit	Theory	Lab
SE 1113	Introduction to Software Engineering	3	3	0

Course Outline: Introduction to Computers, Basic Computer Organization, Processor and Memory,

Secondary Storage Devices, Input-Output Devices, Computer Software, Software and Software Engineering and Software Process Models.

References:

1. Computer Fundamentals, Pradip K Sinha, BPB Publications.
2. Software Engineering: A Practitioner's Approach, 7th Edition, McGraw Hill Higher Education

Year 1 Term 2

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

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.

CSE 1202	Data Structure Lab	2	0	2

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.

MATH 1207	Ordinary Differential Equations	3	3	0

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

Year 2 Term 1

CSE 2101	Algorithm Analysis	2	2	0

Course Code	Course Title	Credit	Theory	Lab

CSE 2101	Algorithm Analysis	2	2	0

Course Outline: Introduction - Algorithms, Analyzing & Designing Algorithms, Correctness of Algorithms; Greedy Algorithms - Introduction to Greedy Algorithms, Greedy Choice Property, Greedy vs. Dynamic Programming, Fractional Knapsack Problem, Activity Selection Problem, Huffman
Encoding, Task Scheduling Problem, Coin Changing Problem, Kruskal's and Prim's Minimum

Spanning Tree Algorithms; Divide and Conquer Algorithms - Introduction to Divide and Conquer Design Technique, Quick Sort, Merge Sort, Proof of Correctness, and Run Time Analysis; Dynamic Programming - Introduction to Dynamic Programming Technique, Principle of Optimality, Optimal Substructure Property, Assembly Line Scheduling, Matrix Chain Multiplication, LCS, Viterbi Algorithm, Bitonic Euclidean Traveling Salesperson Problem and Runtime Analysis; Graph Searching and Shortest Path Problems - Breadth First Search, Depth First Search, Flow Networks, Single Source and All Pair Shortest Path Algorithms; Linear Programming -Overview of Linear Programming, Formulating Problem as Linear Programs, Simplex Algorithm and Integer Linear Programming; Selected Topics - Computational Geometry, Number Theoretic and String Matching Algorithms; NP Completeness and Approximation Algorithms - NP Completeness, Polynomial Time Verification, NP Completeness and Reducibility, NP Complete Problems and Approximation Algorithms.

References:

1. Thomas Corman, *Introduction to Algorithms*, Stein Pub MIT Press, 3rd Ed.
2. Alfred V. Aho, John E. Hopcroft, Jeffrey D. Ullman, *The Design and Analysis of Computer Algorithms*, Addison Wesley Series, 1974 Ed.

Course Code	Course Title	Credit	Theory	Lab

CSE 2102	Algorithm Analysis Lab	1	0	1

Course Outline: Introduction - Algorithms, Analyzing & Designing Algorithms, Correctness of Algorithms; Greedy Algorithms - Introduction to Greedy Algorithms, Greedy Choice Property, Greedy vs. Dynamic Programming, Fractional Knapsack Problem, Activity Selection Problem, Huffman
Encoding, Task Scheduling Problem, Coin Changing Problem, Kruskal's and Prim's Minimum

Spanning Tree Algorithms; Divide and Conquer Algorithms - Introduction to Divide and Conquer Design Technique, Quick Sort, Merge Sort, Proof of Correctness, and Run Time Analysis; Dynamic Programming - Introduction to Dynamic Programming Technique, Principle of Optimality, Optimal Substructure Property, Assembly Line Scheduling, Matrix Chain Multiplication, LCS, Viterbi Algorithm, Bitonic Euclidean Traveling Salesperson Problem and Runtime Analysis; Graph Searching and Shortest Path Problems - Breadth First Search, Depth First Search, Flow Networks, Single Source and All Pair Shortest Path Algorithms; Linear Programming -Overview of Linear Programming, Formulating Problem as Linear Programs, Simplex Algorithm and Integer Linear Programming; Selected Topics - Computational Geometry, Number Theoretic and String Matching Algorithms; NP Completeness and Approximation Algorithms - NP Completeness, Polynomial Time Verification, NP Completeness and Reducibility, NP Complete Problems and Approximation Algorithms.

References:

1. Thomas Corman, *Introduction to Algorithms*, Stein Pub MIT Press, 3rd Ed.
2. Alfred V. Aho, John E. Hopcroft, Jeffrey D. Ullman, *The Design and Analysis of Computer Algorithms*, Addison Wesley Series, 1974 Ed.

Course Code	Course Title	Credit	Theory	Lab
SE 2103	Theory of Computation	2	2	0

Course Outline: Brief Review of mathematical background: Binary relations, digraph, string, languages, proofs, inductive definitions; Finite automata and regular expressions: Deterministic and non-deterministic finite automata, regular expressions and regular sets, Kleene's Theorem; Properties
of regular sets: pumping lemma, closure properties, decision algorithms; Context Free grammar and
languages: Context-free grammars, regular grammars; Simplified forms and normal forms: useful symbols, productions, unit productions, chomsky normal form; Pushdown automata: pushdown automaton, equivalence between pushdown automata and context-free languages; Turing machine: introduction to Turing machines.

References:

1. Introduction to Automata Theory, Languages, and Computation by John E. Hopcroft, Rajeev Motwani, Jeffrey D. Ullman, Third Edition, Pearson Education.

Course Code	Course Title	Credit	Theory	Lab
SE 2104	Theory of Computation Lab	1	0	1

Course Outline: Brief Review of mathematical background: Binary relations, digraph, string,

languages, proofs, inductive definitions; Finite automata and regular expressions: Deterministic and non-deterministic finite automata, regular expressions and regular sets, Kleene's Theorem; Properties of regular sets: pumping lemma, closure properties, decision algorithms; Context Free grammar and languages: Context-free grammars, regular grammars; Simplified forms and normal forms: useful symbols, productions, unit productions, chomsky normal form; Pushdown automata: pushdown automaton, equivalence between pushdown automata and context-free languages; Turing machine: introduction to Turing machines.

References:

1. Introduction to Automata Theory, Languages, and Computation by John E. Hopcroft, Rajeev Motwani, Jeffrey D. Ullman, Third Edition, Pearson Education.

Course Code	Course Title	Credit	Theory	Lab
CSE 2105	Computer Networks	2	2	0

Course Outline: Introduction: Overview of the Internet, Overview of Networking Protocols, Network Edge, Network Core, Protocol Layers / Service Model, General Networking Example; Application Layer: Principles of Networking Applications, Web and HTTP, FTP, E-mail, DNS; Transport Layer: Transport Layer Services, Multiplexing and De multiplexing, Connectionless Transport: UDP, Principles of Reliable Data Transport, Connection-Oriented Transport: TCP, Principles of Congestion Control, TCP Congestion Control; Network Layer: Datagram Networks, Inside a Router, Details of the Internet Protocol (IP), IP Sub netting, Routing Algorithms (Link State, Distance Vector), Routing in the Internet (Routing Information Protocol (RIP), Open Shortest Path First (OSPF), Border Gateway Protocol (BGP)).

References:

1. Data Communications and Networking, B. A. Forouzan, 5/e

Course Code	Course Title	Credit	Theory	Lab
CSE 2106	Computer Networks Lab	1	0	1

Course Outline: Introduction: Overview of the Internet, Overview of Networking Protocols, Network Edge, Network Core, Protocol Layers / Service Model, General Networking Example; Application Layer: Principles of Networking Applications, Web and HTTP, FTP, E-mail, DNS; Transport Layer: Transport Layer Services, Multiplexing and De multiplexing, Connectionless Transport: UDP, Principles of Reliable Data Transport, Connection-Oriented Transport: TCP, Principles of Congestion Control, TCP Congestion Control; Network Layer: Datagram Networks, Inside a Router, Details of the Internet Protocol (IP), IP Sub netting, Routing Algorithms (Link State, Distance Vector), Routing in the Internet (Routing Information Protocol (RIP), Open Shortest Path First (OSPF), Border Gateway Protocol (BGP)).

References:

1. Data Communications and Networking, B. A. Forouzan, 5/e

Course Code	Course Title	Credit	Theory	Lab
MATH 2107	Numerical Analysis for Engineers	2	2	0

Outline: Introductory concepts and calculus review, 'C' programming, the sources and propagation of errors, root finding for nonlinear equations, solution of system of linear equations, interpolation and approximation theory, numerical integration and differentiation.

References:

1. Numerical Methods, E Balagurusamy, Tata McGraw-Hill Publishing Company, 2002

Course Code	Course Title	Credit	Theory	Lab
MATH 2108	Numerical Analysis for Engineers Lab	1	0	1

Outline: Introductory concepts and calculus review, 'C' programming, the sources and propagation of errors, root finding for nonlinear equations, solution of system of linear equations, interpolation and approximation thory, numerical integration and differentiation.

References:

1. Numerical Methods, E Balagurusamy, Tata McGraw-Hill Publishing Company, 2002

Course Code	Course Title	Credit	Theory	Lab
SE 2109	Object Oriented Concepts II	2	2	0

Course Outline: Object Oriented Concepts - Review of Object Oriented Concept - Object Data, Object Behaviors, Class, Attributes, Methods, Encapsulation and Data Hiding: Interfaces and Implementations, Inheritance: Super classes and Sub classes, Abstraction and Is-A relationship, Polymorphism, Compositions: Abstractions and Has-A Relationship; Mastering Composition and Building Objects – Representing Composition with UML, Composition Relationships, Building in Phases, Types of Composition: Aggregation and Associations, Avoiding Dependencies and Cardinality; Details of Creating Object Models with UML – Class Diagram, Attributes and Methods, Access Designations, Inheritance, Interfaces, Composition: Aggregations and Associations, and Cardinality; Objects and Portable Data - Portable Data, The Extensible Markup Language (XML), XML Versus HTML, XML and Object-Oriented Languages, Validating the Document with the Document Type Definition (DTD), Integrating the DTD into the XML Document, and Using Cascading Style Sheets; Persistence objects – Basics of Persistence, Saving to a Flat File, Using XML in the Serialization Process and Writing to a

Relational Database; Objects and the Internet – Object-Based Scripting Languages, Objects in a Web Page and Distributed Objects and the Enterprise; Objects and Client/Server Applications – Client/Server Approaches, Proprietary Approaches and Nonproprietary Approaches; Object Oriented Design Principles - Single Responsibility Principle, Open/Close Principle, Liskov Substitution Principal, Interface Segregation Principle and Dependency Inversion Principle; Introduction to Component Based Design, Design Patterns and Code Smells. **Programming lessons** - 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 – Generics: Wildcard, Generic class definitions, Generic method definitions, Using generics; Collection Framework: Collection interfaces, List and SortedList, Map and SortedMap, Navigable Map, Set and Sorted Set, Navigable Set, Queue and DeQueue, Stack, hashCode() and equals(), Comparator and Comparable; Reflection: The *Class* Class, reflect package, Fields and Methods, Exception Handling and Reflections and Dynamic Programming; Multi-Threaded Programming: Overview of Thread, Java Thread Model, Creating and Running Thread, Thread Pools, Thread Synchronization, wait and notify, join and sleep and The concurrency API; User Interface: Swing, Components, Container, Events, Layouts and SwingWorker; Serialization: Serializable interface, Writing and Reading an Object, Handling Exceptions, Customized Serialization and Controlling Serialization; Socket Programming: Clients and Servers, Ports, Addresses and Protocols, Communication using I/O, Servers, The ServerSocket Class, The URL lass and URLConnction Class; Java Servlet Programming: Introduction To Servlet, Servlet Life cycle, HttpServlet, HttpRequest, HttpResponse, RequestDispatcher, HttpSession and ServletContext, Servlet Configuration, Cookies, Servlet Filters and Http Headers and MIME types; The Java Beans AOI: Introspector, PropertyDescriptor, EventSetDescriptor and MethodDescriptor.

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
4. Head First Java by Kathy Sierra and Bert Bates, O Reilly

Course Code	Course Title	Credit	Theory	Lab
SE 2110	Object Oriented Concepts II Lab	1	0	1

Course Outline: Object Oriented Concepts - Review of Object Oriented Concept - Object Data, Object Behaviors, Class, Attributes, Methods, Encapsulation and Data Hiding: Interfaces and Implementations, Inheritance: Super classes and Sub classes, Abstraction and Is-A relationship, Polymorphism, Compositions: Abstractions and Has-A Relationship; Mastering Composition and Building Objects – Representing Composition with UML, Composition Relationships, Building in Phases, Types of Composition: Aggregation and Associations, Avoiding Dependencies and Cardinality; Details of Creating Object Models with UML – Class Diagram, Attributes and Methods, Access Designations, Inheritance, Interfaces, Composition:

Aggregations and Associations, and Cardinality; Objects and Portable Data - Portable Data, The Extensible Markup Language (XML), XML Versus HTML, XML and Object-Oriented Languages, Validating the Document with the Document Type Definition (DTD), Integrating the DTD into the XML Document, and Using Cascading Style Sheets; Persistence objects – Basics of Persistence, Saving to a Flat File, Using XML in the Serialization Process and Writing to a Relational Database; Objects and the Internet – Object-Based Scripting Languages, Objects in a Web Page and Distributed Objects and the Enterprise; Objects and Client/Server Applications – Client/Server Approaches, Proprietary Approaches and Nonproprietary Approaches; Object Oriented Design Principles - Single Responsibility Principle, Open/Close Principle, Liskov Substitution Principal, Interface Segregation Principle and Dependency Inversion Principle; Introduction to Component Based Design, Design Patterns and Code Smells. **Programming lessons** - 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 – Generics: Wildcard, Generic class definitions, Generic method definitions, Using generics; Collection Framework: Collection interfaces, List and SortedList, Map and SortedMap, Navigable Map, Set and Sorted Set, Navigable Set, Queue and DeQueue, Stack, hashCode() and equals(), Comparator and Comparable; Reflection: The *Class* Class, reflect package, Fields and Methods, Exception Handling and Reflections and Dynamic Programming; Multi-Threaded Programming: Overview of Thread, Java Thread Model, Creating and Running Thread, Thread Pools, Thread Synchronization, wait and notify, join and sleep and The concurrency API; User Interface: Swing, Components, Container, Events, Layouts and SwingWorker; Serialization: Serializable interface, Writing and Reading an Object, Handling Exceptions, Customized Serialization and Controlling Serialization; Socket Programming: Clients and Servers, Ports, Addresses and Protocols, Communication using I/O, Servers, The ServerSocket Class, The URL lass and URLConnction Class; Java Servlet Programming: Introduction To Servlet, Servlet Life cycle, HttpServlet, HttpRequest, HttpResponse, RequestDispatcher, HttpSession and ServletContext, Servlet Configuration, Cookies, Servlet Filters and Http Headers and MIME types; The Java Beans AOI: Introspector, PropertyDescriptor, EventSetDescriptor and MethodDescriptor.

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
4. Head First Java by Kathy Sierra and Bert Bates, O Reilly

Course Code	Course Title	Credit	Theory	Lab
SE 2112	Software project Lab I	3	0	3

Course Outline: Each of the students should complete the software project separately. They will be marked based on their individual software. Student will be encouraged to develop software which requires significant “problem solving” effort. The project should be sufficiently large and the size of the project will mostly depend on “problem solving” effort. Besides, students must showcase the skills they have acquired from their so far completed courses.

Year 2 Term 2

Course Code	Course Title	Credit	Theory	Lab
CSE 2201	Operating Systems and System Programming	2	2	0

Course Outline: Introduction: What is operating system? History of operating system Operating system concepts Operating system structure Processes and Threads Processes Threads Interprocess communication Scheduling Classical IPC problems Memory Management No memory abstraction Virtual memory Page replacement algorithms Design issues for paging systems Implementation issues File Systems Files Directories File system management Input / Ouptut Principles of I/O hardware Principles of I/O software I/O software layers Disks Clocks Thin clients Deadlocks Resources Detection Recovery Avoidance Prevention Virtualization and Cloud

Course Reference Books:

1. Operating System Concepts, 7th edition, by Silberschatz, Galvin, Gagne
2. Modern Operating Systems, 4th edition, Tanenbum, Bos

Course Code	Course Title	Credit	Theory	Lab
CSE 2202	Operating Systems and System Programming Lab	1	0	1

Course Outline: Introduction: What is operating system? History of operating system Operating system concepts Operating system structure Processes and Threads Processes Threads Interprocess communication Scheduling Classical IPC problems Memory Management No memory abstraction Virtual memory Page replacement algorithms Design issues for paging systems Implementation issues File Systems Files Directories File system management Input / Ouptut Principles of I/O hardware Principles of I/O software I/O software layers Disks Clocks Thin clients Deadlocks Resources Detection Recovery Avoidance Prevention Virtualization and Cloud

Course Reference Books:

1. Operating System Concepts, 7th edition, by Silberschatz, Galvin, Gagne
2. Modern Operating Systems, 4th edition, Tanenbum, Bos

Course Code	Course Title	Credit	Theory	Lab
GE 2203	Business Psychology	3	3	0

Course Outline: Fundamentals: Definition of Psychology, Subfields of Psychology, Major Perspectives of Psychology, Psychology in Business; Job Analysis: Job-oriented Approach, Person-oriented Approach, Purposes of Job Analysis, Methods of Job Analysis, Job Evaluation; Assessment Methods for Selection and Placement: Psychological Tests: Ability Test, Personality Test, Intelligence Test, Vocational Interest Test; Training and Development: Training Need Analysis, Training Designs, Training Methods, Evaluation of Training; Theories of Employee Motivation: Need Theories, Reinforcement Theory, Expectancy Theory, Goal Setting Theory; Job Attitude and Emotion: Nature of Job Satisfaction, Assessment of Job Satisfaction, Antecedents of Job Satisfaction, Potential Effects of Job Satisfaction, Organizational Commitment, Emotion at work; Productive and Counterproductive Employee Behavior: Productive Behavior, Job Performance; Counterproductive Behavior, Withdrawal, Aggression, Mistreatment, Sabotage, and Theft; Occupational Health Psychology: Occupational Health and Safety, Work Schedules, Occupational Stress, Work-Family Conflict, Burnout, Hawthorne Studies; Leadership: Approaches to the Understanding of Leadership Trait Approach, Leader Behavior Approach, Contingency Theory, Path-Goal Theory, Leader-Member Exchange (LMX) Theory, Transformational Leadership Theory; Organizational Development and Theory: Organizational Development Employee Acceptance of Change, Management by Objectives, Survey Feedback, Team Building, T-Group; Effectiveness of Organizational Development: Organizational Theories, Bureaucracy, Theory X and Theory Y, Open System Theory, Socio-technical System Theory.

References:

3. Industrial and Organizational Psychology: Research and Practice, Paul E. Spector, 5th Edition

Course Code	Course Title	Credit	Theory	Lab
CSE 2205	Information Security	2	2	0

Course Outline: Overview: Network Security Concepts, Security Attacks, Services and Mechanisms; Classical Encryption techniques: Symmetric Cipher Model, Substitution and Permutation Ciphers, Steganography; Block Ciphers and Data Encryption Standard: Design principles and modes of operation; Public-key cryptography: Introduction to number theory, RSA and Diffie-Hellman; Message Digest: Requirements for cryptographic hash functions,MD5 ,SHA, Message authentication codes, digital signatures; Key Management and Distribution: Symmetric Key Distribution using Symmetric Encryption, Symmetric Key Distribution using asymmetric Encryption, public key distribution, public key certificates, x.509 certificates; Network and Internet Security: Transport Layer Security, Wireless LAN security, e-mail security.

References:

1. Information Security: Principles and Practice by Mark Stamp 2nd Edition Wiley 2011

Course Code	Course Title	Credit	Theory	Lab
CSE 2206	Information Security Lab	1	0	1

Course Outline: Overview: Network Security Concepts, Security Attacks, Services and Mechanisms; Classical Encryption techniques: Symmetric Cipher Model, Substitution and Permutation Ciphers, Steganography; Block Ciphers and Data Encryption Standard: Design principles and modes of operation; Public-key cryptography: Introduction to number theory, RSA and Diffie-Hellman; Message Digest: Requirements for cryptographic hash functions,MD5 ,SHA, Message authentication codes, digital signatures; Key Management and Distribution: Symmetric Key Distribution using Symmetric Encryption, Symmetric Key Distribution using asymmetric Encryption, public key distribution, public key certificates, x.509 certificates; Network and Internet Security: Transport Layer Security, Wireless LAN security, e-mail security.

References:

1. Information Security: Principles and Practice by Mark Stamp 2nd Edition Wiley 2011

Course Code	Course Title	Credit	Theory	Lab
CSE 2207	Database Management System-I	2	2	0

Course Outline: Introduction to Database Systems: Evolution of file processing systems, role of databases in organizations, core components of a database environment; Data Modeling: the Entity-Relationship Diagram and its symbols and constructs; The Relational Model and Normalization: relational model, normalization, transformation of an entity-relationship data diagram into a relational model; SQL - A Standard Navigation Language for Relational Databases; Overview of Object-Oriented Databases: object-oriented data model, implementation of object persistence using relational databases.

References:

1. Database System Concepts by Avi Silberschatz, Henry F. Korth and S. Sudarshan, Sixth Edition

Course Code	Course Title	Credit	Theory	Lab
CSE 2208	Database Management System-I Lab	1	0	1

Course Outline: Introduction to Database Systems: Evolution of file processing systems, role of databases in organizations, core components of a database environment; Data Modeling: the Entity-Relationship Diagram and its symbols and constructs; The Relational Model and Normalization: relational model, normalization, transformation of an entity-relationship data diagram into a relational model; SQL - A Standard Navigation Language for Relational Databases; Overview of Object-Oriented Databases: object-oriented data model, implementation of object persistence using relational databases.

References:

1. Database System Concepts by Avi Silberschatz, Henry F. Korth and S. Sudarshan, Sixth Edition

Course Code	Course Title	Credit	Theory	Lab
SE 2209	Software Requirements Spec. and Analysis	2	2	0

Course Outline: Review of – The Nature of Software, Software Engineering, The Software Process, Software Engineering Practices, Generic Software Process Model, Process Assessment and Improvement, Prescriptive Process Models, Specialized Process Model and Agile Development. Requirements Engineering, Establishing the ground work, Eliciting Requirements, Negotiating Requirements, Validating Requirements, Requirements Analysis, Scenario-Based Modeling, UML Models, Data Modeling Concept, Class Based Modeling, Requirements Modeling Strategies, Flow-Oriented Model, Behavioral Model, Requirements Modeling for WebApps.

References:

1. R. S. Pressman, Software Engineering. A Practitioner's Approach, 7/e or higher, McGraw Hill
2. Ian Sommerville. Software Engineering, 9th or higher Edition, Addison-Wesley.

Course Code	Course Title	Credit	Theory	Lab
SE 2210	Software Requirements Spec. and Analysis Lab	1	0	1

Lab: One small real life system will be given to all the students for analyzing in the class room. Three real life mid-scale systems will be distributed among groups (created randomly) of 5/6 students to analyze (one project per group). The output of both of the analysis will be specification reports.

Course Code	Course Title	Credit	Theory	Lab
BUS 2211	Business Studies for Engineers	3	3	0

Course Outline: Managers and Entrepreneurs: Management Defined, Role of a Manager, Small-Business Management, The Evolution of Management Thought, Organization, Organization Charts, Contrasting Theories of Organization, Organizational Effectiveness, Organizational Cultures, Change, Conflict, and Negotiation in Organization; The Strategic Management Process, Strategic Implementation and Control, Forecasting. **Accounting Basic:** Forms of Business Organization, Types of Activities performed by Business Organization, Financial statements of Business Organization, The Accounting Equation, The Account and Rules of Debit and Credit, The Journal: Recording of Transaction, Adjusting the Accounts, Closing Entries, and Preparing Financial statements from the Work Sheet. **Analysis and Interpretation of Financial Statement:** Objectives of Financial Statement Analysis, Analysis of a Balance Sheet, Analysis of Statement of Income and Retained Earnings, Ratio Analysis: Liquidity Ratios, Equity or Long Term Solvency Ratio, Profitability Test, Market Test.

References:

1. Stephen P. Robbins and Mary Coulter, *Management*, Prentice Hall, Latest Edition
2. Jerry J. Weygandt, Donald E. Kieso, and Paul D. Kimmel, *Accounting Principles*, Wiley, 8th Ed.

Year 3 Term 1

Course Code	Course Title	Credit	Theory	Lab
SE 3101	Professional Ethics for Information Systems	3	3	0

Course Outline: Introduction to Ethics, Morals, Integrity, Ethical use of Information Technology, Ethics for IT Workers and IT Users - Trade secret, Whistle blowing, fraud, misrepresentation, bribery, professional code of ethics, IT professional malpractice and Common Ethical issues for IT users, Computer and Internet Crime - Exploit, Viruses, Phishing and Types of perpetrators, Privacy - Information privacy, fair information practices, EU data protection directive, key privacy and anonymity issues, Freedom of Expression - Right to freedom of expression, obscene speech, hate speech, defamation, controlling access to information on the internet, anonymity on the internet, corporate blogging and pornography, Intellectual property - Copyright, fair use doctrine, patent, software patents, trade secret and key intellectual property issues, Impact of IT on society, Social networking ethical issues, Ethics for IT organization.

References:

1. Ethics In Information Technology, George W. Reynolds

Course Code	Course Title	Credit	Theory	Lab
CSE 3103	Web Technology	1	1	0

Course Outline: Introduction To Html, Java Script & CSS, Server Side Programming: HTTP Server, Application Server, MVC Web Framework, Web Services, Database Access: Object Relational Mapping, Lambda Expression, Language Integrated Query, Data Reader, Writer, Web Security: Denial of Service, Buffer Overflow, Cross Site Scripting, Authentication and Access Control

References:

1. Deitel & Deitel, Goldberg, “Internet and world wide web – How to Program”, Pearson
   Education Asia, 2001.
2. Rajkamal, “Web Technology”, Tata McGraw-Hill, 2001.
3. Teach yourself web technologies part I & II- I. Bayross. BPB
4. Web Design in a Nutshell- J. Niederst, SPD

Course Code	Course Title	Credit	Theory	Lab
CSE 3104	Web Technology Lab	2	0	2

Course Outline: Introduction To Html, Java Script & CSS, Server Side Programming: HTTP Server, Application Server, MVC Web Framework, Web Services, Database Access: Object Relational Mapping, Lambda Expression, Language Integrated Query, Data Reader, Writer, Web Security: Denial of Service, Buffer Overflow, Cross Site Scripting, Authentication and Access Control

References:

1. Deitel & Deitel, Goldberg, "Internet and world wide web – How to Program", Pearson
   Education Asia, 2001.
2. Rajkamal, "Web Technology", Tata McGraw-Hill, 2001.
3. Teach yourself web technologies part I & II- I. Bayross. BPB
4. Web Design in a Nutshell- J. Niederst, SPD

Course Code	Course Title	Credit	Theory	Lab
CSE 3105	Data Science and Analytics – DBMS II	1	1	0

Course Outline: Indexing and Hashing: Basic Concepts, Ordered Indices, B+-Tree Index Files, B-Tree Index Files, Static Hashing, Dynamic Hashing, Comparison of Ordered Indexing and Hashing; Query Processing: Overview, Measures of Query Cost, Selection Operation, Sorting, Join Operation, Evaluation of Expressions; Query Optimization: Introduction, Transformation of Relational Expressions, Catalog Information for Cost Estimation, Statistical Information for Cost Estimation, Cost-based optimization; Transactions: Transaction Concept, Transaction State, Concurrent Executions, Serializability; Concurrency Control: Lock-Based Protocols, Timestamp-Based Protocols; Recovery System: Failure Classification, Storage Structure, Recovery and Atomicity, Log-Based Recovery, Recovery With Concurrent Transactions; Data Analysis and Mining: Data Mining, Decision tree, Bayes theory, Randomize tree; Database System Architectures: Centralized and Client-Server Systems, Server System Architectures, Parallel Systems, Distributed Systems, Network Types; Parallel Databases: Introduction, I/O Parallelism, Interquery Parallelism, Intraquery Parallelism, Intraoperation Parallelism, Interoperation Parallelism; Distributed Databases: Heterogeneous and Homogeneous Databases: Distributed Data Storage, Distributed Transactions, Commit Protocols; Additional should be included: Database Design, Database Tuning Security and Authorization, Multidimensional

query.

References:

1. Ramez Elmasri and Shamkant B. Navathe Fundamentals of Database Systems. Third Edition. Addison-Wesley Pub Co, 1999.
2. Database Systems: The Complete Book, Hector Garcia-Molina, Jeffrey D. Ullman and Jennifer D. Widom Prentice Hall. (best supporting book)
3. Fundamentals of Database Systems, by Ramez Elmasri and Shamkant Navathe, Addison Wesley.
4. Database System Concepts,Fifth Edition, Avi Silberschatz, Henry F. Korth, S. Sudarshan (text book)

Course Code	Course Title	Credit	Theory	Lab
CSE 3106	Data Science and Analytics – DBMS II Lab	2	0	2

Course Outline: Indexing and Hashing: Basic Concepts, Ordered Indices, B+-Tree Index Files, B-Tree Index Files, Static Hashing, Dynamic Hashing, Comparison of Ordered Indexing and Hashing; Query Processing: Overview, Measures of Query Cost, Selection Operation, Sorting, Join Operation, Evaluation of Expressions; Query Optimization: Introduction, Transformation of Relational Expressions, Catalog Information for Cost Estimation, Statistical Information for Cost Estimation, Cost-based optimization; Transactions: Transaction Concept, Transaction State, Concurrent Executions, Serializability; Concurrency Control: Lock-Based Protocols, Timestamp-Based Protocols; Recovery System: Failure Classification, Storage Structure, Recovery and Atomicity, Log-Based Recovery, Recovery With Concurrent Transactions; Data Analysis and Mining: Data Mining, Decision tree, Bayes theory, Randomize tree; Database System Architectures: Centralized and Client-Server Systems, Server System Architectures, Parallel Systems, Distributed Systems, Network Types; Parallel Databases: Introduction, I/O Parallelism, Interquery Parallelism, Intraquery Parallelism, Intraoperation Parallelism, Interoperation Parallelism; Distributed Databases: Heterogeneous and Homogeneous Databases: Distributed Data Storage, Distributed Transactions, Commit Protocols; Additional should be included: Database Design, Database Tuning Security and Authorization, Multidimensional query.

References:

1. Ramez Elmasri and Shamkant B. Navathe Fundamentals of Database Systems. Third Edition. Addison-Wesley Pub Co, 1999.
2. Database Systems: The Complete Book, Hector Garcia-Molina, Jeffrey D. Ullman and Jennifer D. Widom Prentice Hall. (best supporting book)
3. Fundamentals of Database Systems, by Ramez Elmasri and Shamkant Navathe,

Addison Wesley.

4. Database System Concepts,Fifth Edition, Avi Silberschatz, Henry F. Korth, S. Sudarshan (text book)

Course Code	Course Title	Credit	Theory	Lab
BUS 3107	Business Communications	2	2	0

Course Outline: Communication Concept: The Role of Communication in Business, Importance of Communication Skills, Main Form of Business Communication, Process of Human Communication. **Fundamentals of Business Writing:** Adaptation and the Selection of Words, Construction of Clear Sentences and Paragraphs, Writing for Effect. **Basic Pattern of Business Messages:** Directness in Good News and Neutral Situations, Indirectness in Bad Message, Indirectness in Persuasion Message, Letter and Memorandum, Letter Writing Styles, Pattern Variations in Memorandums and the Email, Job Search Activities: Strategies in the Job Search Process, Job search activities, Writing CV, Facing Interviews, Feedback letters for goodwill, **Fundamentals of Report Writing:** Basics of Report Writing, Report Structure: The Short Forms, Long and Formal Report, Usages of Graphics. **Other Form of Business Communication:** Informal Oral Communication, Technology-Enabled Communication.

References:

1. Raymond V. Lesikar, John D. Pettit, Maire E. Flatley, Lesikar's Basic Business Communication, Mc Graw Hill.

Course Code	Course Title	Credit	Theory	Lab
BUS 3108	Business Communications Lab	1	0	1

Course Outline: Communication Concept: The Role of Communication in Business, Importance of Communication Skills, Main Form of Business Communication, Process of Human Communication. **Fundamentals of Business Writing:** Adaptation and the Selection of Words, Construction of Clear Sentences and Paragraphs, Writing for Effect. **Basic Pattern of Business Messages:** Directness in Good News and Neutral Situations, Indirectness in Bad Message, Indirectness in Persuasion Message, Letter and Memorandum, Letter Writing Styles, Pattern Variations in Memorandums and the Email, Job Search Activities: Strategies in the Job Search Process, Job search activities, Writing CV, Facing Interviews, Feedback letters for goodwill, **Fundamentals of Report Writing:** Basics of Report Writing, Report Structure: The

Short Forms, Long and Formal Report, Usages of Graphics. **Other Form of Business Communication:** Informal Oral Communication, Technology-Enabled Communication.

References:

1. Raymond V. Lesikar, John D. Pettit, Maire E. Flatley, Lesikar's Basic Business Communication, Mc Graw Hill.

Course Code	Course Title	Credit	Theory	Lab
SE 3109	Design Pattern	2	2	0

Course Outline: Revision of Concepts of OOP, Importance of learning design patterns, Types of Design Patterns - Structural, Behavioral and Creational Patterns, Creational Patterns – Singleton, Factory, Factory Method, Abstract Factory, Builder, Prototype and Object Pool, Behavioral Patterns - Chain of Responsibility, Command, Interpreter, Iterator, Mediator, Memento, Observer, Strategy, Template Method, Visitor and Null Object, Structural Patterns – Adapter, Bridge, Composite, Decorator, Flyweight and Proxy, REFACTORING CODE SMELL, Different type of code smells - Inappropriate Naming, Comments, Dead Code, Duplicated code, Primitive Obsession, Large Class, Lazy Class, Alternative Class with Different Interface, Long Method, Long Parameter List, Switch Statements, Speculative Generality, Oddball Solution, Feature Envy, Refused Bequest, Black Sheep and Train Wreck, Design Principles (SOLID) - Single responsibility principle, Open Close Principle, Liskov substitution principle, Interface segregation principle, Dependency Inversion principle.

References:

1. Gamma, Erich. *Design patterns: elements of reusable object-oriented software*. Pearson Education, 1995.

Course Code	Course Title	Credit	Theory	Lab
SE 3110	Design Pattern Lab	1	0	1

Course Outline: Revision of Concepts of OOP, Importance of learning design patterns, Types of Design Patterns - Structural, Behavioral and Creational Patterns, Creational Patterns – Singleton, Factory, Factory Method, Abstract Factory, Builder, Prototype and Object Pool, Behavioral Patterns - Chain of Responsibility, Command, Interpreter, Iterator, Mediator, Memento, Observer, Strategy, Template Method, Visitor and Null Object, Structural Patterns – Adapter, Bridge, Composite, Decorator, Flyweight and Proxy, REFACTORING CODE SMELL, Different type of code smells - Inappropriate Naming, Comments, Dead Code, Duplicated code, Primitive Obsession, Large Class, Lazy Class, Alternative Class with Different Interface, Long

Method, Long Parameter List, Switch Statements, Speculative Generality, Oddball Solution, Feature Envy, Refused Bequest, Black Sheep and Train Wreck, Design Principles (SOLID) - Single responsibility principle, Open Close Principle, Liskov substitution principle, Interface segregation principle, Dependency Inversion principle.

References:

1. Gamma, Erich. *Design patterns: elements of reusable object-oriented software*. Pearson Education, 1995.

Course Code	Course Title	Credit	Theory	Lab
SE 3112	Software Project Lab II	3	0	3

Course Outline: Students will create project teams of 3 members each. Number of team members can be varied for special cases, decided by the assigned course manager. All the project teams are required to prepare their Software Requirements Specification (SRS) first, and later develop the project accordingly.

Year 3 Term 2

Course Code	Course Title	Credit	Theory	Lab
CSE 3201	Distributed Systems	1	1	0

Course Outline: Foundations - Characterization of DS, System Models, Networking and Internetworking, Interprocess Communication, Remote Invocation, Indirect Communication and Operating System Support **Middleware** - Dist. Objects and Components, Web Services and Peer-to-Peer Systems System services – Security, Distributed File Systems and Name Services Distributed algorithms - Time and Global States, Coordination and Agreement Shared data, Transactions and Concurrency Control, Distributed Transactions, and Replication, New challenges -Mobile and Ubiquitous Computing

References:

1. Distributed Systems: Concepts and Design (5th Edition). George Coulouris (Author), Jean Dollimore (Author), Tim Kindberg (Author), Gordon Blair (Author)

Course Code	Course Title	Credit	Theory	Lab
CSE 3202	Distributed Systems Lab	2	0	2

Lab: Introduction to Message passing technology and its applications, Sockets Programming, Remote Procedure Calls code implementation, Synchronization assignments, Group Communication code implementation, Distributed mutual exclusion assignment, Implementation of Election Algorithms, Implementation of Distributed File system: MapReduce, Spanner, Distributed Systems Design assignments: Cloud Services and Content Delivery Networks configuration.

Course Code	Course Title	Credit	Theory	Lab
SE 3203	Software Metrics	2	2	0

Course Outline: Overview of Software Metrics, The basics of Measurement, Goal based framework for software measurement, Empirical Investigation, Measuring Internal Attributes : Size, Measuring Internal Attributes : Structure, Measuring Cost and Effort, Measuring External product attributes : Quality, Measuring Software Reliability, Object Oriented Metrics, For hands-on experiences: Students will implement different software metrics calculation related algorithms, utilize existing industry related tools for measuring software metrics and compare it with their implementations to gain concrete idea.

References:

1. Software metrics- A Rigorous and Practical Approach, (3rd Edition) Norman Fenton, and Jones

Bieman.

2. Software Measurement and Estimation: A practical Approach (1st Edition) Linda M. Laird, and M. Carol Brennan

Course Code	Course Title	Credit	Theory	Lab
SE 3204	Software Metrics Lab	1	0	1

Course Outline: Overview of Software Metrics, The basics of Measurement, Goal based framework for software measurement, Empirical Investigation, Measuring Internal Attributes : Size, Measuring Internal Attributes : Structure, Measuring Cost and Effort, Measuring External product attributes : Quality, Measuring Software Reliability, Object Oriented Metrics, For hands-on experiences: Students will implement different software metrics calculation related algorithms, utilize existing industry related tools for measuring software metrics and compare it with their implementations to gain concrete idea.

References:

1. Software metrics- A Rigorous and Practical Approach, (3rd Edition) Norman Fenton, and Jones Bieman.
2. Software Measurement and Estimation: A practical Approach (1st Edition) Linda M. Laird, and M. Carol Brennan

Course Code	Course Title	Credit	Theory	Lab
SE 3205	Software Security	2	2	0

Course Outline: Introduction: Security principles, concept of computer security, security services and policies Security risks: Database security, operating systems security, secure coding Countermeasures: methodologies and tools for identifying and eliminating security vulnerabilities, techniques to prove the absence of vulnerabilities, and ways to avoid security holes in new software. Secure software design: essential guidelines for building secure software, information security standards

Suggested Readings:

1. Security in Computing, 4th Edition, by Charles P. Pfleeger , Publisher: Prentice Hall; 4th edition
2. Computer security: principles and practices, by William Stallings and Lawrie Brown, 2nd Edition,
3. Brian Chess and Jacob West, Secure Programming with Static Analysis (required)
4. David A. Wheeler, Secure Programming for Linux and Unix HOWTO Version 3.5, Aug 2004 (required)
5. Goertzel et al, Software Security Assurance State of the Art Report, May 2007.
6. Aleph One, Smashing the Stack for Fun and Profit. Phrack Vol 7, Nr. 49

7. Tim Newsham, Format String Attacks, Guardent tech report, Sept 2000

Course Code	Course Title	Credit	Theory	Lab
SE 3206	Software Security Lab	1	0	1

Course Outline: Introduction: Security principles, concept of computer security, security services and policies Security risks: Database security, operating systems security, secure coding Countermeasures: methodologies and tools for identifying and eliminating security vulnerabilities, techniques to prove the absence of vulnerabilities, and ways to avoid security holes in new software. Secure software design: essential guidelines for building secure software, information security standards

Suggested Readings:

1. Security in Computing, 4th Edition, by Charles P. Pfleeger , Publisher: Prentice Hall; 4th edition
2. Computer security: principles and practices, by William Stallings and Lawrie Brown, 2nd Edition,
3. Brian Chess and Jacob West, Secure Programming with Static Analysis (required)
4. David A. Wheeler, Secure Programming for Linux and Unix HOWTO Version 3.5, Aug 2004 (required)
5. Goertzel et al, Software Security Assurance State of the Art Report, May 2007.
6. Aleph One, Smashing the Stack for Fun and Profit. Phrack Vol 7, Nr. 49
7. Tim Newsham, Format String Attacks, Guardent tech report, Sept 2000

Course Code	Course Title	Credit	Theory	Lab
CSE 3207	Artificial Intelligence	2	2	0

Course Outline: Intelligent Agents and their Environments - The concept of a Rational Agent, Specifying the Task environment (PEAS description), Different characteristics of environments (Fully vs Partially observable, Static vs Dynamic, Episodic vs Sequential etc.) and Different types of agents (Reflex, Goal-based, Utility-based etc.),Search - Formulating a search problem , Uninformed Search strategies: BFS, DFS, DLS, ID-DFS, their working principles, complexities, relative advantages and disadvantages, Informed (heuristic) Search strategies: Greedy Best-first search, A* search: Working principle, Characteristics of heuristics (admissibility and consistency), Proof of A*'s optimality, Local search: Hill Climbing, Searching with non-deterministic actions: AND-OR search trees and Searching with partial observability: Belief state-space search, Adversarial Search - Formulation of a Game tree,

The minimax algorithm, Alpha-Beta pruning: Its rationale, working principle and Additional techniques such as Move ordering and Search cut-off, Probabilistic Reasoning - Bayes' rule and its uses, Bayesian Network: Building a Bayes-net and making inference from it, Markov Chains and Hidden Markov Models: Transition and Sensor models, Building and HMM, applications of HMM, Inference in temporal models: Filtering, Prediction, Most Likely explanations (Viterbi algorithm) etc. and Particle Filters: basic working principle, Making Decisions - Decision theory and Utility theory: Lottery, Utility functions, Maximum Expected Utility principle, Constraints of Utility (Orderability, Transitivity etc) and Markov Decision Processes: Policies, Rewards, Optimal policies and the Utility of States, Value Iteration, Supervised Learning - Basic concepts of classification and supervised learning: Training set, Test set, Overfitting, Underfitting etc., Decision trees: Basic understanding, Learning a Decision tree through entropy calculation, Nearest Neighbor classifier: Basic working principle, Relative advantages and disadvantages, Naive Bayes classifier: Basic working principle, Calculating classification procedures, Relative advantages and disadvantages, Artificial Neural Network: Basic working principle, Basic structure and calculation of a perceptron, Basics of backpropagation algorithm and Support Vector Machines: Basic working principle, Unsupervised Learning (Clustering) - Basic concepts and applications of Clustering, Different types of Clustering: Partitional vs. Hierarchical, Exclusive vs Overlapping vs Fuzzy, Complete vs Partial, K-means Clustering: Basic working principle, characteristics, advantages, disadvantages, Agglomerative Hierarchical Clustering: Basic concepts, Representations (Dendrograms and Nested cluster diagrams), Different techniques to define cluster proximity: Single link, Complete link, Group average, Centroid method, their relative advantages and disadvantages and DBSCAN: Basic principle and applications, Classification of points (Core, Border and Noise), Reinforcement Learning - Understanding basics of Reinforcement Learning: MDPs, Policies, Rewards, Utilities etc., Passive and Active Reinforcement Learning, Exploration and Exploitation, Adaptive Dynamic Programming, Temporal Difference Learning and Q-Learning.

References:

1. Russell, Stuart, and Peter Norvig. "Artificial intelligence: a modern approach." (1995).

Course Code	Course Title	Credit	Theory	Lab
CSE 3208	Artificial Intelligence Lab	1	0	1

Course Outline: Intelligent Agents and their Environments - The concept of a Rational Agent, Specifying the Task environment (PEAS description), Different characteristics of environments (Fully vs Partially observable, Static vs Dynamic, Episodic vs Sequential etc.) and Different types of agents (Reflex, Goal-based, Utility-based etc.),Search - Formulating a search problem , Uninformed Search strategies: BFS, DFS, DLS, ID-DFS, their working principles, complexities, relative advantages and disadvantages, Informed (heuristic) Search strategies: Greedy Best-first search, A* search: Working principle, Characteristics of heuristics (admissibility and consistency), Proof of A*'s optimality, Local search: Hill Climbing, Searching with non-deterministic actions: AND-OR search trees and Searching with partial observability: Belief state-space search, Adversarial Search - Formulation of a Game tree, The minimax algorithm, Alpha-Beta pruning: Its rationale, working principle and Additional techniques such as Move ordering and Search cut-off, Probabilistic Reasoning - Bayes' rule and its uses, Bayesian Network: Building a Bayes-net and making inference from it, Markov Chains and Hidden Markov Models: Transition and Sensor models, Building and HMM, applications of HMM,

Inference in temporal models: Filtering, Prediction, Most Likely explanations (Viterbi algorithm) etc. and Particle Filters: basic working principle, Making Decisions - Decision theory and Utility theory: Lottery, Utility functions, Maximum Expected Utility principle, Constraints of Utility (Orderability, Transitivity etc) and Markov Decision Processes: Policies, Rewards, Optimal policies and the Utility of States, Value Iteration, Supervised Learning - Basic concepts of classification and supervised learning: Training set, Test set, Overfitting, Underfitting etc., Decision trees: Basic understanding, Learning a Decision tree through entropy calculation, Nearest Neighbor classifier: Basic working principle, Relative advantages and disadvantages, Naive Bayes classifier: Basic working principle, Calculating classification procedures, Relative advantages and disadvantages, Artificial Neural Network: Basic working principle, Basic structure and calculation of a perceptron, Basics of backpropagation algorithm and Support Vector Machines: Basic working principle, Unsupervised Learning (Clustering) - Basic concepts and applications of Clustering, Different types of Clustering: Partitional vs. Hierarchical, Exclusive vs Overlapping vs Fuzzy, Complete vs Partial, K-means Clustering: Basic working principle, characteristics, advantages, disadvantages, Agglomerative Hierarchical Clustering: Basic concepts, Representations (Dendrograms and Nested cluster diagrams), Different techniques to define cluster proximity: Single link, Complete link, Group average, Centroid method, their relative advantages and disadvantages and DBSCAN: Basic principle and applications, Classification of points (Core, Border and Noise), Reinforcement Learning - Understanding basics of Reinforcement Learning: MDPs, Policies, Rewards, Utilities etc., Passive and Active Reinforcement Learning, Exploration and Exploitation, Adaptive Dynamic Programming, Temporal Difference Learning and Q-Learning.

References:

1. Russell, Stuart, and Peter Norvig. "Artificial intelligence: a modern approach." (1995).

Course Code	Course Title	Credit	Theory	Lab
SE 3209	Software Testing and Quality Assurance	2	2	0

Course Outline: The Psychology and Economics of Software Testing, Software Testing Life Cycle (STLC), Software Testing Terminology and Methodology, V&V Model, Dynamic Black Box Testing – Boundary Value Analysis, Equivalence Partitioning, State Transition based Testing, Decision Table based Testing, Cause-Effect Graphing based Testing and Error Guessing, Dynamic White Box Testing – Basis Path Testing, Data Flow Testing and Mutation Testing, Inspections, Walkthroughs, Technical Reviews, Unit Testing, Integration Testing, Function Testing, System Testing, Acceptance Testing, Regression Testing, Test Management – Test Organization, Test Plan, Test Design and Specifications, Software Metrics, Software Quality, Quality Control and Quality Assurance, Quality Management and Project Management, Software Quality Metrics, Testing Internet Applications - Security and Performance Testing, Debugging, Test Driven Development (TDD), Behavior Driven Development (BDD). **Tools and Project** - The students will be divided into small groups having at most 3 members and a class project will be given to them for preparing a system test case. They must validate the requirements and create Mock UIs during the preparation of test cases. Besides, each of the students will relate their learnings on unit, regression, performance and security testing, debugging, behavior driven development via different tools like JUnit, Selenium, Apache JMeter, Sprajax, Sqlninja, Bugzilla, Cucumber

References:

1. Naresh Chauhan, Software Testing: Principles and Practices, 1st or higher Edition, Oxford University Press.
2. Glenford J. Myers, Corey Sandler, and Tom Badgett. The Art of Software Testing, 3rd or higher Edition, John Wiley & Sons.
2. Lisa Crispin and Janet Gregory. Agile Testing: A Practical Guide for Testers and Agile Teams, 1st or higher Edition, Pearson Education.

Course Code	Course Title	Credit	Theory	Lab
SE 3210	Software Testing and Quality Assurance Lab	1	0	1

Course Outline: The Psychology and Economics of Software Testing, Software Testing Life Cycle (STLC), Software Testing Terminology and Methodology, V&V Model, Dynamic Black Box Testing – Boundary Value Analysis, Equivalence Partitioning, State Transition based Testing, Decision Table based Testing, Cause-Effect Graphing based Testing and Error Guessing, Dynamic White Box Testing – Basis Path Testing, Data Flow Testing and Mutation Testing, Inspections, Walkthroughs, Technical Reviews, Unit Testing, Integration Testing, Function Testing, System Testing, Acceptance Testing, Regression Testing, Test Management – Test Organization, Test Plan, Test Design and Specifications, Software Metrics, Software Quality, Quality Control and Quality Assurance, Quality Management and Project Management, Software Quality Metrics, Testing Internet Applications - Security and Performance Testing, Debugging, Test Driven Development (TDD), Behavior Driven Development (BDD). **Tools and Project** - The students will be divided into small groups having at most 3 members and a class project will be given to them for preparing a system test case. They must validate the requirements and create Mock UIs during the preparation of test cases. Besides, each of the students will relate their learnings on unit, regression, performance and security testing, debugging, behavior driven development via different tools like JUnit, Selenium, Apache JMeter, Sprajax, Sqlninja, Bugzilla, Cucumber

References:

1. Naresh Chauhan, Software Testing: Principles and Practices, 1st or higher Edition, Oxford University Press.
2. Glenford J. Myers, Corey Sandler, and Tom Badgett. The Art of Software Testing, 3rd or higher Edition, John Wiley & Sons.
3. Lisa Crispin and Janet Gregory. Agile Testing: A Practical Guide for Testers and Agile Teams, 1st or higher Edition, Pearson Education.

Course Code	Course Title	Credit	Theory	Lab
SE 3211	Software Design and Architecture	2	2	0

Course Outline: Design Concept - The Design Process, Design Concepts, The Design Model; Architectural Design: Software Architecture, Architectural Genres, Architectural Styles, Architectural Design, Assessing, Alternative Architectural Designs, Architectural Mapping Using Data Flow; Component-Level Design: What Is a Component, Designing Class-Based Components, Conducting Component-Level Design, Component-Level Design for WebApps, Designing Traditional Components, Component-Based Development; User Interface Design: The Golden Rules, User Interface Analysis and Design, Interface Analysis, Interface Design Steps, Web App Interface Design, Design Evaluation.

References:

1. Software Engineering – A Practitioner's Approach. 7th Edition, Roger S. Pressman
2. Software Engineering. 9th Edition, Ian Sommerville

Course Code	Course Title	Credit	Theory	Lab
SE 3212	Software Design and Architecture Lab	1	0	1

Course Outline: Design Concept - The Design Process, Design Concepts, The Design Model; Architectural Design: Software Architecture, Architectural Genres, Architectural Styles, Architectural Design, Assessing, Alternative Architectural Designs, Architectural Mapping Using Data Flow; Component-Level Design: What Is a Component, Designing Class-Based Components, Conducting Component-Level Design, Component-Level Design for WebApps, Designing Traditional Components, Component-Based Development; User Interface Design: The Golden Rules, User Interface Analysis and Design, Interface Analysis, Interface Design Steps, Web App Interface Design, Design Evaluation.

References:

1. Software Engineering – A Practitioner's Approach. 7th Edition, Roger S. Pressman
2. Software Engineering. 9th Edition, Ian Sommerville

Year 4 Term 1

Course Code	Course Title	Credit	Theory	Lab
SE 4100	Internship	18	0	18

Outline:
The student will work full-time as an intern to particular company for a period of six months. S/he will be evaluated based on the marks provided by the company along with the marks of at least two presentations given at IIT.

Year 4 Term 2

Course Code	Course Title	Credit	Theory	Lab
SE 4202	Project	6	0	6

Course Outline: Each student can perform a software development or research project. For a research project a student has to submit a thesis. For software development project, a student should submit documents having the following: Project proposal, Software Requirements Specification, Software Design Specification, Software Test Plan and User Manual. Besides, each of the students has to give multiple intermediate presentations to report their project progress.

Course Code	Course Title	Credit	Theory	Lab
SE 4203	Software Maintenance	2	2	0

Course Outline: Lifecycle roadmap is presented and different types of evolution and maintenance activities are placed on it. Status within industry and research is mapped out. Evolution laws are discussed and analyses in different contexts (the traditional, component-based, and open-source). Pre delivery and transition maintenance models are studied and criteria for their success are identified. Impact analysis is studied. Different ways to manage customer problems are surveyed, both the critical (emergency) and non-critical ones. Retirement process model is considered. Finally, the quality attribute "maintainability" is discussed and compared to "bad smells". Techniques for evolutionary design in the small: refactoring. Techniques for large-scale evolutionary design, especially evolution of legacy systems. Architectural patterns for isolation/exposure of change.

References:

1. Effective Software Maintenance and Evolution: A Reuse-Based Approach by Stanislaw Jarzabek; Publisher Taylor & Francis
2. Software Maintenance: Concepts and Practice By Penny Grubb, Armstrong A. Takang 2nd edition World Scientific USA.

Course Code	Course Title	Credit	Theory	Lab
SE 4204	Software Maintenance Lab	1	0	1

Course Outline: Lifecycle roadmap is presented and different types of evolution and maintenance activities are placed on it. Status within industry and research is mapped out. Evolution laws are discussed and analyses in different contexts (the traditional, component-based, and open-source). Pre delivery and transition maintenance models are studied and criteria for their success are identified.

Impact analysis is studied. Different ways to manage customer problems are surveyed, both the critical (emergency) and non-critical ones. Retirement process model is considered. Finally, the quality attribute "maintainability" is discussed and compared to "bad smells". Techniques for evolutionary design in the small: refactoring. Techniques for large-scale evolutionary design, especially evolution of legacy systems. Architectural patterns for isolation/exposure of change.

References:

1. Effective Software Maintenance and Evolution: A Reuse-Based Approach by Stanislaw Jarzabek; Publisher Taylor & Francis
2. Software Maintenance: Concepts and Practice By Penny Grubb, Armstrong A. Takang 2nd edition World Scientific USA.

Course Code	Course Title	Credit	Theory	Lab
SE 4205	Software Project Management	2	2	0

Course Outline: Introduction to Project management: Historical background and evolution, Terminologies, Software project management objectives, Scope, focus and basic rules Principles of software project management: Basic PM Skills, SPM framework, elements, stakeholders, boundaries, challenges of SPM Software Project planning: Planning objective, project plan, variations, structure of SPM plan, project estimation, estimation methods, models and decision process. PM organization and scheduling: WBS, types of WBS, functions, activities, tasks, life cycles, phasing and purpose of phasing, building project schedule, network diagrams: PERT, CPM, Bar charts, Gantt charts Software project management techniques: Use of methodologies, Managing risks and issues, Managing Quality, Configuration, Change, Crisis, Documentation, Release. Project monitoring and control: Dimensions of monitoring and control, earned value indicators (BCWS, CV, SV, CPI, SPI), backlog management, dispute and error tracking, RMMM charts Industry scenarios: Domain analysis, Business case analysis, Dynamicity, Success and failure factors, case studies

References:

1. Stellman, Andrew, and Jennifer Greene. *Applied software project management*. " O'Reilly Media, Inc.", 2005.
2. Phillips, Joseph. *IT project management: on track from start to finish*. McGraw-Hill, Inc., 2002.
3. Rubin, Kenneth S. *Essential Scrum: A practical guide to the most popular Agile process*. Addison-Wesley, 2012.

Course Code	Course Title	Credit	Theory	Lab
SE 4206	Software Project Management Lab	1	0	1

Course Outline: Introduction to Project management: Historical background and evolution, Terminologies, Software project management objectives, Scope, focus and basic rules Principles of software project management: Basic PM Skills, SPM framework, elements, stakeholders, boundaries, challenges of SPM Software Project planning: Planning objective, project plan, variations, structure of SPM plan, project estimation, estimation methods, models and decision process. PM organization and scheduling: WBS, types of WBS, functions, activities, tasks, life cycles, phasing and purpose of phasing, building project schedule, network diagrams: PERT, CPM, Bar charts, Gantt charts Software project management techniques: Use of methodologies, Managing risks and issues, Managing Quality, Configuration, Change, Crisis, Documentation, Release. Project monitoring and control: Dimensions of monitoring and control, earned value indicators (BCWS, CV, SV, CPI, SPI), backlog management, dispute and error tracking, RMMM charts Industry scenarios: Domain analysis, Business case analysis, Dynamicity, Success and failure factors, case studies

References:

1. Stellman, Andrew, and Jennifer Greene. *Applied software project management*. " O'Reilly Media, Inc.", 2005.
2. Phillips, Joseph. *IT project management: on track from start to finish*. McGraw-Hill, Inc., 2002.
3. Rubin, Kenneth S. *Essential Scrum: A practical guide to the most popular Agile process*. Addison-Wesley, 2012.

Elective Courses

Course Code	Course Title	Credit	Theory	Lab
CSE 804	Information Retrieval	3	2	1

Course Outline: Boolean Retrieval: Inverted Index, Processing boolean queries, extended Boolean retrieval; Term Vocabulary and Postings lists: Document delineation and character sequence decoding, Tokenization, Dropping common terms: stop words, Normalization (equivalence classing of terms), Stemming and lemmatization, skip pointers, Biword indexes, Positional indexes; Dictionaries and tolerant retrieval: Search structures for dictionaries, General wildcard queries, k-gram indexes for wildcard queries, Spelling correction; Index Construction: Blocked sort-based indexing, Single-pass in-memory indexing, Distributed indexing, Dynamic indexing; Scoring and Ranking: Parametric and zone indexes, Term frequency and weighting, The vector space model for scoring, variant tf-idf functions; Computing scores in a complete search system: Efficient scoring and ranking, Components of an information retrieval system; Evaluation in information retrieval: Evaluation of unranked retrieval sets, Evaluation of ranked retrieval results, Assessing relevance, Results snippets; Relevance feedback and query expansion: The Rocchio algorithm for relevance feedback, Relevance feedback on the web, Evaluation of relevance feedback strategies, Global methods for query reformulation; Language models for information retrieval; Enterprise Information Retrieval: Explore the capacity of Apache Lucene as a text search framework.

References:

1. An Introduction to Information Retrieval by Christopher D. Manning, Prabhakar Raghavan, Hinrich Schütze, Online Edition, 2009, Cambridge University Press, Cambridge, England.
2. Lucene in Action by Michael McCandless, Erik Hatcher, and Otis Gospodnetić, Second Edition, Manning publications.

Course Code	Course Title	Credit	Theory	Lab
CSE 825	Data Mining and Warehousing	3	2	1

Course Outline: Introduction to Data Mining, Knowing Data (Data objects, similarities and dissimilarities, statistical descriptions and visualizations), Data Pre-processing, Data Warehousing and Online Analytical Processing, Data Cube technology, Mining frequent patterns, Classification and Cluster Analysis, Research trends in Data mining and warehousing.

References:

1. Data Mining: Concepts and Techniques. Jiawei Han, Micheline Kambar, Jian Pei [Text Book]

Course Code	Course Title	Credit	Theory	Lab
CSE 829	Pattern Recognition and Image Processing	3	2	1

Course Outline: Introduction to Image Processing; Digital Image Fundamentals - Elements of Visual Perception. Light and the Electromagnetic Spectrum. Image Sensing and Acquisition. Image Sampling and Quantization. Some Basic Relationships between Pixels. Linear and Nonlinear Operations; Image Enhancement in the Spatial Domain - Background. Some Basic Gray Level Transformations. Histogram Processing. Enhancement Using Arithmetic/Logic Operations. Basics of Spatial Filtering. Smoothing Spatial Filters. Sharpening Spatial Filters. Combining Spatial Enhancement Methods; Image Enhancement in the Frequency Domain - Background. Introduction to the Fourier Transform and the Frequency Domain. Smoothing Frequency-Domain Filters. Sharpening Frequency Domain Filters. Homomorphic Filtering. Implementation; Image Restoration - A Model of the Image Degradation/Restoration Process. Noise Models. Restoration in the Presence of Noise Only-Spatial Filtering. Periodic Noise Reduction by Frequency Domain Filtering. Linear, Position-Invariant Degradations. Estimating the Degradation Function. Inverse Filtering. Minimum Mean Square Error (Wiener) Filtering. Constrained Least Squares Filtering. Geometric Mean Filter. Geometric Transformations; Color Image Processing - Color Fundamentals. Color Models. Pseudo color Image Processing. Basics of Full-Color Image Processing. Color Transformations. Smoothing and Sharpening. Color Segmentation. Noise in Color Images. Color Image Compression; Wavelets and Multiresolution Processing - Background. Multiresolution Expansions. Wavelet Transforms in One Dimension. The Fast Wavelet Transform. Wavelet Transforms in Two Dimensions. Wavelet Packets; Image Compression - Fundamentals. Image Compression Models. Elements of Information Theory. Error-Free Compression. Lossy Compression. Image Compression Standards; Morphological Image Processing - Preliminaries. Dilation and Erosion. Opening and Closing. The Hit-or-Miss Transformation. Some Basic Morphological Algorithms. Extensions to Gray-Scale Images; Image Segmentation - Detection of Discontinuities. Edge Linking and Boundary Detection. Thresholding. Region-Based Segmentation. Segmentation by Morphological Watersheds. The Use of Motion in Segmentation; Representation and Description - Representation. Boundary Descriptors. Regional Descriptors. Use of Principal Components for Description. Relational Descriptors; Object Recognition - Patterns and Pattern Classes. Recognition Based on Decision-Theoretic Methods. Structural Methods.

References:

1. Digital Image Processing - Rafael C Gonzalez and Richard E. Woods

Course Code	Course Title	Credit	Theory	Lab
CSE 831	Computer Graphics and Multimedia	3	2	1

Course Outline: Introduction: History of computer graphics, graphics architectures and software, imaging: pinhole camera, human vision, synthetic camera, modeling vs. rendering OpenGL: architecture, displaying simple two-dimensional geometric objects, positioning systems, working in a windowed environment Color: Color perception, color models (RGB, CMY, and HLS), color transformations. Color in OpenGL. RGB and Indexed color. Input: working in a network environment, client-server computing; input measure, event, sample and request input, using callbacks, picking.

Geometric transformations: affine transformations (translation, rotation, scaling, and shear), homogeneous coordinates, concatenation, current transformation and matrix stacks. Three dimensional graphics: classical three dimensional viewing, specifying views, affine transformation in3D, projective transformations. Ray Tracing. Shading: illumination and surface modeling, Phong shading model, polygon shading. Rasterization: line drawing via Bresenham's algorithm, clipping, polygonal fill, BitBlt. Introduction to hidden surface removal (z buffer). Discrete Techniques: buffers, bitblt, reading and writing bitmaps and pixel maps, texture mapping, compositing.

References:

1. Computer Graphics, Principle and Practices – James D. Foley, Andries van Dam, Steven K. Feiner and John F. Hughes.

Course Code	Course Title	Credit	Theory	Lab
BUS 842	Strategic Management	3	3	0

Course Outline: Strategic Management Concept: Strategic Leadership, Competitive Advantage, Superior Performance, Performance in Nonprofit Enterprises, Strategic Manager, Strategy Making Process. **Industry analysis, External Environment and Internal Resources Analysis:** Industry and Sector, market segments, Porter's Five Forces Model, , Strategic Groups, Industry Life Cycle Analysis, Macroeconomic Forces, Competitive Advantage, Value Creation and Profitability, Avoiding Failures and Sustaining Competitive Advantage. **Functional and Business Level Strategy:** Achieving Superior Efficiency, Learning Effects, Materials Management, Strategy for Attaining superior Reliability, Responsiveness to Customers, Competitive Positioning and Business Level Strategy, Strategies in Fragmented Industries, Embryonic, Growth and Mature Industries. **Technological Support for Adopting Strategies and Global Strategy:** Format War, Strategies for wining in Format War, Information System Strategy, Managing Intellectual Property Rights, Capturing First-Mover Advantages, Technological Paradigm Shifts, Disruptive Technology, Profitability and Profit Growth through Global Expansion, Global Standardization Strategy. **Strategic Software Engineering:** Architecture-Centric Software Development Strategy, Software Product Lines, Software Effort and Cost Estimation Strategies, Openness of a Software, Software Supply Chain, Software Economics.

References:

1. Theory of Strategic Management (Eighth Edition) By: Hill/Jones
2. Strategic Management (Concepts and Cases) Twelfth Edition By: Fred R. David

Course Code	Course Title	Credit	Theory	Lab
EEE 202	Digital System Design	3	2	1

Course Outline:

- **Introduction:** Number System, Number Base Conversation, Complements, Signed Number. Arithmetic Operation- Binary, Octal, Hexadecimal Binary Codes e.g. BCD, ASCII, Grey etc.
- **Boolean Algebra:** Theorems & Properties of Boolean Algebra, Boolean Functions, Canonical and Standard Forms and Simplification.
- **Logic Gates:** Switching Circuits, Electronic Logic Gates, Gate Symbols, Design and operation of NOT, OR, AND, NOR, NAND, XOR, XNOR Gates. Analysis of Combinational Circuits- Algebraic Method, Truth Table Method.
- **Synthesis of Combinational Logic Circuits:** AND-OR NAND Networks, OR-AND and NOR Network, AND-OR-Invert Circuits.
- **Combinational Logic Design:** Circuits (gate level), Design Hierarchy and procedures. Two-level and multi-level implementations, Arithmetic operation using gates (add, subtract, multiply), Logic Minimization, K-Map, Unate Covering, Quine McCluskey Method, CAD tools for two level minimization, ESPRESSO Algorithm and other popular (multiplexers, encoders, decoders) modules design.
- **Programmable Logic Devices:** Technologies, Performance, Classical and Mid-Complexity Architectures (PLDs, CPLDs, FPGAs) and Modern Architectures (SoPC).
- **Sequential Logic Design:** Latches, Flip-Plops, State Machine Design & Minimization (Mealy and Moore models) and Design Problems.
- Sequential Circuits: Design of Synchronous Counters, Ripple counters, parallel Load counters, Introduction of Registers and shift Register:
- **Memory Design:** Random Access Memory (RAMS), Static RAMS, Dynamic RAMS, Memory organizations and Read only Memories (ROM)

References:

1. Digital Logic Circuit Analysis and Design by Vicor P. Nelson and H. Troy Nagle, Bill D. Carroll, J. David Irwin.
2. Logic and Computer Design Fundamentals by M.M. Mano and C.R. Kime, Prentice-Hall, 4th Ed.
3. Introduction to Digital Logic Design by J.P. Hayes, Addison-Wesley, 1993.
4. Digital Systems: Principles and Applications, by Ronal J Tocci, Neal Widmer, Gregory L Moss, Prentice-Hall 1997.
5. Fundamentals of Digital Logic with VHDL Design by S. Brown and Z Vranesic, McGraw-Hill, 2nd Ed.
6. Analysis and Design of Digital Systems with VHDL by Allen Dewey, PWS Publication, 1st Ed.

Course Code	Course Title	Credit	Theory	Lab
CSE 303	Data Communication and Networking	3	2	1

Outline: Introduction: Overview of Data communication, networking and network models; Physical Layer and Media: Data and Signal, Digital Transmission, Analog Transmission, Transmission Media Data Link Layer: Error detection and Correction, Flow and error control, Medium access control protocols (ALOHA, CSMA/CD, CSMA/CA), Channelization (FDMA, TDMA, CDMA) Ethernet, Wireless LANs; Network Layer: Logical Addressing.

References:

1. Data Communications and Networking, B. A. Forouzan, 5/e

Course Code	Course Title	Credit	Theory	Lab
CSE 501	Parallel Computing	3	2	1

Course Outline: Introduction, Parallel Computer Model, the State of Computing, Flynn's Classification, Parallel/Vectors Computers, and Challenges for Parallel Computing, System Attributes to Performance, Clock Rate and CPI, Performance Factors, System Attributes, MIPS rates, Throughput rate, Implicit and Explicit Parallelism, Multiprocessors and Multicomputers, Shared Memory Multiprocessors, different types of model (UMA, NUMA, COMA), Distributed Memory Multicomputer, Multicomputer Generation, Multivector and SIMD Computers, Vector supercomputer, SIMD supercomputers, PRAM and VLSI Models, Parallel Random Access Machines, Time and Space Complexities, NP Completeness, PRAM models, VLSI complexity model and discussion about related papers, Introduction to Program and Network Properties, Condition of Parallelism, Data and Resource Dependencies, Control Dependence, Resource Dependence, Bernstein's Conditions, Hardware and Software Parallelism, The role of compiler, Program Partitioning and Scheduling, Grain Sizes and Latencies, Grain Packing and Scheduling, Static Multiprocessor Scheduling, Node duplication, Program Flow Mechanisms, Control Flow versus Data Flaw, Demand Driven Mechanisms, System Interconnect Architectures, Network Properties and Routing, Node Degree and Network Diameter, Bisection Width, Data Routing Function (Permutations, Perfect Shuffle and Exchange, Hypercube Routing Functions, Broadcast and Multi cast), Network Performance, Related Paper Studies, Static connection Networks (Linear array, Ring and Chordal Ring, Barrel Shifter, Tree and Star, Mesh and Torus, Systolic Array, Hypercube, Cube Connected Cycles), Network Throughput, Dynamic Connection Networks, Principles of Scalable Performance, Performance Metrics and Measures, Parallelism Profile in Programs, Degree of Parallelism, Average Parallelism, Available Parallelism, Asymptotic Speedup, Harmonic Mean Performance, Arithmetic Mean Performance, Geometric mean Performance, harmonic Mean Performance, Harmonic Mean Speedup, Amdahl's Law, System Efficiency, Quality of Parallelism, Scalability of Parallel Algorithms, and Speedup Performance Laws, Processors and Memory Hierarchy: Advanced processor Technology, Design Space of Processors, Instruction Pipelines, Processors and Coprocessors, Superscalar and Vector Processors, Virtual Memory Technology, Pipelining and Superscalar Techniques: Linear Pipeline Processors, Asynchronous and Synchronous Models, Speedup Efficiency and Throughput, Nonlinear Pipeline

Processors, Reservation and Latency Analysis, Collision Free Scheduling, state diagram, greedy cycles, Pipeline Schedule Optimization, Pipeline Throughput, and Instruction Pipeline Design, Branch Handling Techniques, Effects of Branch, Related Paper discussion, Multiprocessors and Multicomputer: Hierarchical Bus Systems, Crossbar Switch and Multi port Memory, Hot spot problem, Cache Coherence and Synchronization Mechanisms, cache coherence problem, process migration, snoopy bus protocols, Directory based protocols, Hardware Synchronizations Mechanisms, Software for Parallel Programming: Object Oriented Model, Functional and Logic Model, Parallel Language and Compilers.

References:

1. Kai Hwang, "Advanced Computer Architecture", McGraw-Hill.
2. Principles of Parallel Programming, by Calvin Lin and Larry Snyder, Addison-Wesley, 2009.
3. Patterns for Parallel Programming, by Mattson, Sanders, and Massingill, Addison-Wesley, 2005.
4. The Art of Multiprocessor Programming, by Herlihy and Shavit, Morgan Kaufmann, 2008.

Course Code	Course Title	Credit	Theory	Lab
BUS 602	Management Information Systems	3	1	2

Course Outline: Foundation Concepts a) Foundations of Information Systems in Business i. Foundation Concepts: Information Systems in Business ii. Foundation Concepts: The components of Information Systems b) Competing with Information Technology i. Fundamentals of Strategic Advantage ii. Using Information Technology for Strategic Advantage 2. Information Technologies a) Computer Hardware b) Computer Software c) Data Resource Management i. Technical Foundations of Database Management ii. Managing Data Resources d) Telecommunications and Networks i. The Networked Enterprise ii. Telecommunications Network Alternatives 3. Business Applications a) E-business Systems i. e-Business Systems ii. Functional Business Systems b) Enterprise Business Systems i. Management at Enterprise Level ii. Enterprise Resource Planning iii. Supply Chain Management c) E-commerce Systems i. e-Commerce Fundamentals ii. e-Commerce Applications and Issues d) Supporting Decision Making i. Decision Support in Business ii. Artificial Intelligence Technologies in Business 4. Development Process a) Developing Business / IT Strategies i. Planning Fundamentals ii. Implementation Challenges b) Developing Business / IT solutions i. Developing Business Systems ii. Implementing Business Systems 5. Management Challenges a) Security and Ethical Challenges i. Security, Ethical and Societal Challenges of IT ii. Security Management of Information Technology b) Enterprise and Global Management of Information Technology i. Managing Information Technology ii. Managing Global IT

Reference:

1. Management Information Systems, 10th edition, James O' Brien

Course Code	Course Title	Credit	Theory	Lab
CSE 802	Computer, Data and Network Security	3	1	2

Course Outline: Overview: Network Security Concepts, Security Attacks, Services and Mechanisms; Classical Encryption techniques: Symmetric Cipher Model, Substitution and Permutation Ciphers, Steganography; Block Ciphers and Data Encryption Standard: Design principles and modes of operation; Public-key cryptography: Introduction to number theory, RSA and Diffie-Hellman; Message Digest: Requirements for cryptographic hash functions,MD5 ,SHA, Message authentication codes, digital signatures; Key Management and Distribution: Symmetric Key Distribution using Symmetric Encryption, Symmetric Key Distribution using asymmetric Encryption, public key distribution, public key certificates, x.509 certificates; Network and Internet Security: Transport Layer Security, Wireless LAN security, e-mail security.

References:

1. Data and Computer Communications By Stallings, 8th Edition, Pearson Education, 2007

Course Code	Course Title	Credit	Theory	Lab
CSE 844	Applied Data Science	3	2	1

Course Outline:

Theory: Introduction to applied data science, Data cleaning/Data Publishing, Data visualization techniques, Predictive analytics, Bayesian analytics, Building efficient models from complex data, Regularization, Opportunities involving applied data science. The course will cover techniques for collecting, storing, and analyzing data in varying formats. Scientific programming, supervised and unsupervised analytics and data visualization techniques will be covered. Topics will involve learning classifiers, Bayesian, maximum a posteriori, parameter estimation, decision trees, neural networks, support vector machines, bag of words classifiers, N-gram models, association rules, nearest neighbor classifiers, locally weighted regression, ensemble classifiers, Clustering, mixture models, k-means clustering, hierarchical clustering, distributional clustering, selected applications in data mining, automated knowledge acquisition, pattern recognition, program synthesis, text and language processing, internet-based information systems, etc.

Lab: The course lab aims to provide an introduction to various topics such as Big Data, Pattern Discovery, Data Visualization, along with a toolkit to use with data i.e., Hadoop, TensorFlow, etc..

Text Book:

Mount and Zumel (2014), Practical data science with R.

References:

1. Cathy O'Neil and Rachel Schutt, *Doing Data Science, O'Reilly, 2014*
2. Russell Jurney, *Agile Data Science, O'Reilly, 2013.*
3. Edward Tufte, *The Visual Display of Quantitative Information, Graphics Press, 2013 (2nd ed).*
4. Morgan Kaufmann, *Data Mining: Practical Machine Learning Tools and Techniques. 3 edition, 2011*
5. Matthew Russell, *Mining the Social Web: Data Mining Facebook, Twitter, LinkedIn, Google+, GitHub, and More. O'Reilly, 2013*

Rules and Regulations of Lab Usage

Institute of Information Technology (IIT), Noakhali Science and Technology University is an educational institution aimed at developing efficient human resource in the field of information and communication technology. IIT currently offers Bachelor of Science and Software Engineering (BSSE) is a four-year program, based on the industry-oriented trend and Post Graduate Diploma in Information Technology (PGDIT). Bachelor of Science in Software Engineering (BSSE) is a four-year program, based on the industry-oriented trend and culture. To operate the above programs, IIT has a well-furnished computer lab. The lab contains 50 computers with a server and 24 hour supervised by CCTV camera. The lab is wifi-enable and each computer is connected to the internet. To use the lab, each students has to obey and maintain the following set of rules & regulations.

1. Keep own shoes at outside of the lab.
2. Keep the lab neat & use bins for any rubbish.
3. Smoking and\or eating in the lab can damage the equipment and attract insects so these are not allowing in the lab.
4. Students should maintain professional and polite communication with the lab supervisor. No awful, hostile or aggressive behavior will be tolerated.
5. Behavior and activities that disturb other students or disrupt the operations of the lab are not allowed.
6. Electronic devices should be used on a professional level. If any students don't understand or has any confusion about the usage of any electrical device than he\she has to ask help to the lab supervisor.
7. Students must have to inform the lab supervisor of any problems that arise at the time of using computer equipment.
8. Students cannot store any personal file like personal images, songs in the lab computer.
9. Students have to log out computers wherever he/she out of his/her desk and shut down the computers when he/she leaves the lab.
10. Students could not modify any software or files or overwrite the operating system, modify the autoexec.bat or config.sys or any other system parameters.

11. Computers and peripherals are not to be moved or reconfigured without approval of lab supervisor or teacher.
12. For any hardware, software, a student should have to contact with lab supervisor or the respective teacher.
13. As the lab is air conditioned, so doors and windows must be kept closed.

A breach of any of the above regulations will constitute a breach of discipline and will be subject to the appropriate disciplinary procedures.