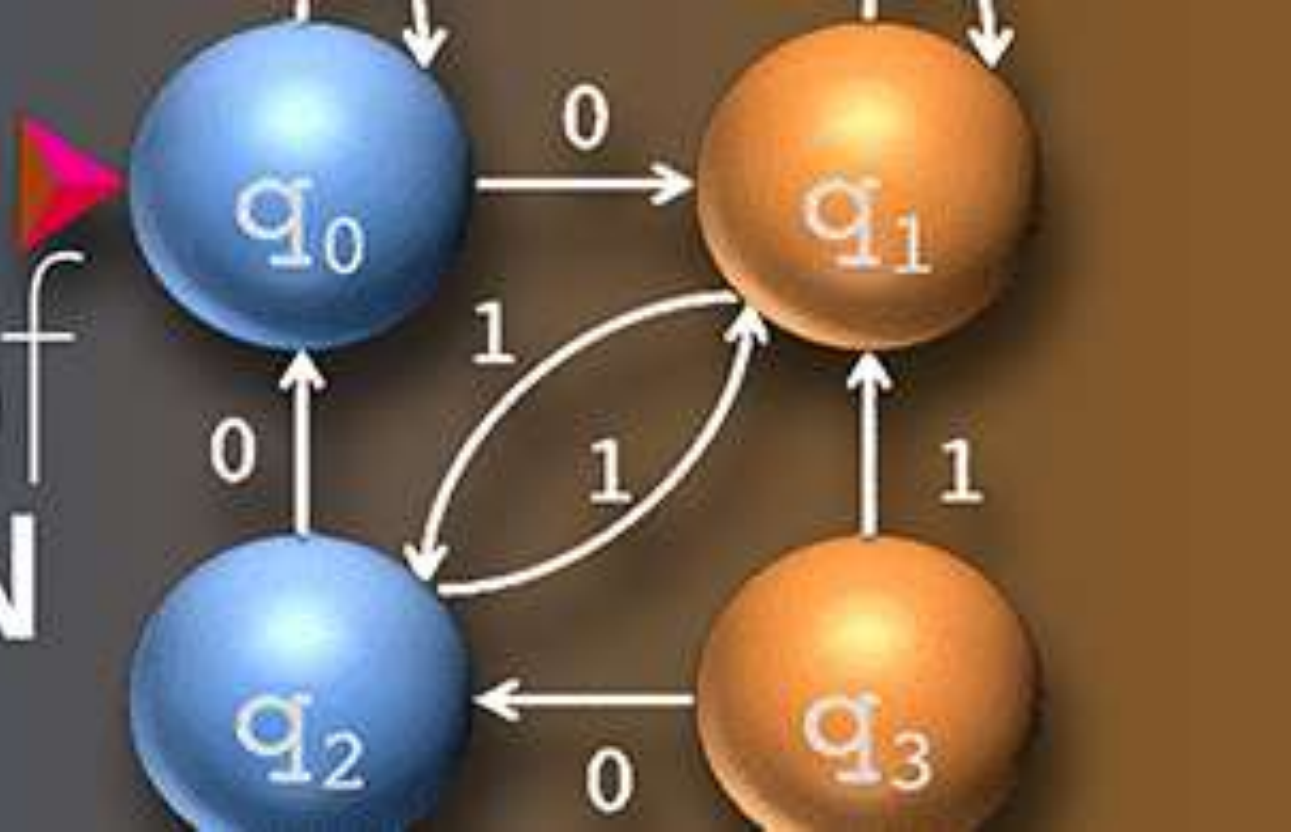
CSE 305

Theory of COMPUTATION



Lecture 1

Course Overview

Md. Mijanur Rahman, Prof. Dr.

Dept. of Computer Science and Engineering,
Jatiya Kabi Kazi Nazrul Islam University, Bangladesh.
www.mijanrahman.com

Contents

Course Overview

- About the course
- Purpose and Motivation
- What is the “Theory of Computation”?
- Why Study Theory of computation
- Course Contents
- Learning Outcomes
- List of Reference Books

This course

- This is an introductory course (theory course) on **Theory of Computation** intended for undergraduate students in Computer Science and Engineering.
- In this course we will introduce various **models of computation and study their power and limitations**.
- This course focuses the three central areas of Theory of Computation:
  - **Automata, Computability, and Complexity**.
  - They are linked by the question: **“What are the fundamental capabilities and limitations of computers?”**

This course

- **PRE-REQUISITES:**
  - It is recommended that the candidate has done a course in **Data Structures and Algorithms**.
- We will start with **Automata Theory**, followed by **Computability Theory**.
- This course widens your understanding of computers and will **influence other courses including Compilers, Programming Languages, and Artificial Intelligence**.

Class Duration

- The duration of each semester will be 19 weeks whose breakdown is as follows:

Class	14 weeks
Recess before Semester Final Examination	2 weeks
Semester Final Examination	3 weeks
Total	19 weeks

- For theoretical course (Credit 3.0), three hours classes will be conducted in a week.

Course Evaluation

- **Evaluation System for theoretical course -** The marking and student evaluation system will be as follows:

1. **Continuous Assessment : 40%**
   a. Attendance : 10%
   b. Mid Semester Exam-1 : 10%
   c. Mid Semester Exam-2 : 10%
   d. Mid Semester Exam-3 : 10%
2. **Semester Final Exam : 60%**

Class Attendance

- The distribution of marks for class attendance (theoretical and practical) will be as follows:

Attendance	Marks
90% and above	10
85% to 89%	09
80% to 84 %	08
75% to 79 %	07
70% to 74%	06
65% to 69%	05
60 % to 64%	04
55% to 59 %	03
50% to 54%	02
Less than 50%	00

- A student shall have to attend at least 75% of theoretical and practical classes held in a course. In case of shortage of attendance (not bellow 60%), student will be allowed to sit for examination after paying of taka 500/- as irregular fee for each course in university account.
- **Below 60% will NOT be allowed to sit for examination.**

The Grading System

- Letter grades and corresponding grade points will be awarded in accordance with the provisions shown below:

Numerical Grade	Letter grade	Grade Point	Interpretation
80% and above	A+	4.00	**Outstanding**
75% to less than 80%	A	3.75	**Excellent**
70% to less than 75%	A-	3.50	**Very Good**
65% to less than 70%	B+	3.25	**Good**
60% to less than 65%	B	3.00	**Satisfactory**
55% to less than 60	B-	2.75	**Nearly Satisfactory**
50% to less than 55%	C+	2.50	**Average**
45% to less than 50%	C	2.25	**Nearly Average**
40% to less than 45%	D	2.00	**Poor**
Less than 40%	**F**	**0**	**Fail**

Purpose and Motivation

- This course is on the Theory of Computation, which tries to answer the following questions:
  - What are the mathematical properties of computer hardware and software?
  - What is a computation and what is an algorithm? Can we give rigorous mathematical definitions of these notions?
  - What are the limitations of computers? **Can "everything" be computed?**

Purpose and Motivation

- The main purpose of theory of computation is to **develop a formal mathematical model of computation that reflects the real world computers.**
- This field of research was started by mathematicians and logicians in the 1930's, when they were trying to understand the meaning of a "computation". **A central question asked was whether all mathematical problems can be solved in a systematic way.**
- Nowadays, the Theory of Computation can be divided into the following three areas: **Complexity Theory, Computability Theory, and Automata Theory.**

What is the "Theory of Computation"?

- Basic question: **What can or cannot be computed with given resources:** primitive operations, bounds on resources (time, memory=space, wire length, fanout, energy consumed, ..)
- The **Theory of Computation** is a branch of computer science and mathematics that **focuses on determining problems that can be solved mechanically, using an algorithm or a set of programming rules.** It is also concerned with the efficiency at which the algorithm can perform the solution.
- **Content of the course:** Automata and Formal Languages, Computability, and Complexity Theory.

What is the "Theory of Computation"?

- In simple terms, the Theory of Computation answers these questions:
  - What problems can the machine solve? What problems can't it solve?
  - How fast can a machine solve a problem?
  - How much memory space does a machine need to solve a problem?
- To answer these questions, **computer scientists use a model of computation, which is a computer simulation for the algorithm being developed**. The Turing machine is among the most used models of computation.

Why Study Theory of computation

- The major reasons about the importance to study of theory of computation are listed below;
  1. This course is about rigorously analyzing capabilities and limitations of computers or real-time systems. **To rectify the limitations of computers and answer what kind of problems can be computed?** These topics form the core of computer science.
  2. **To achieve deep understanding about the mathematical properties of computer hardware and software.** This theory simplifies the complex computers to an abstract and simple mathematical model, and helps us to understand them better.
  3. **To achieve knowledge about mathematical definitions of the computation and the algorithm.** Thus, it is important to study the theory of computation is to better understand the development of formal mathematical models of computation that reflect the real-world of computer.
  4. This theory is very much relevant to practice, for example, in **the design of new programming languages, compilers, string searching, pattern matching, computer security, artificial intelligence, etc.** This course also helps us to learn problem solving skills. Theory teaches us how to think, prove, argue, solve problems, express, and abstract.

Course Contents

CSE 305	Theory of Computation	Cr. 3.0

Course Syllabus:

Introduction to theory of computation: Automata, Computability, and Complexity; Languages and problems; finite automata and regular expressions; properties of regular sets, pumping lemma, closure properties, decision algorithms and Myhill-Nerode theorem; context-free grammars; push down automata; context free languages; Turing machines, Chomsky hierarchy, Recursive, Valid and Invalid computation, Deterministic context-free languages.

Course Contents

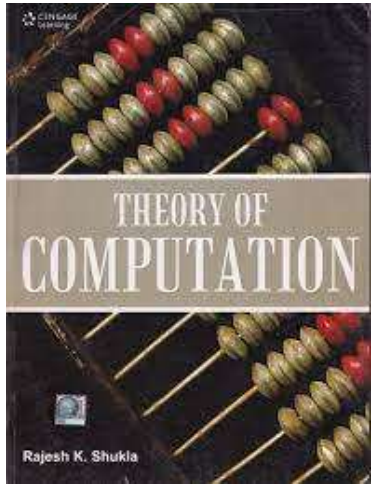
- Major Topics Covered in This Course:
  - Mathematical preliminaries
  - Problems and proof techniques
  - Fundamentals of three central areas: automata theory, computability theory, and computational complexity theory
  - Finite automata and regular languages
  - Context-free grammars and languages
  - Pushdown automata
  - Turing machines

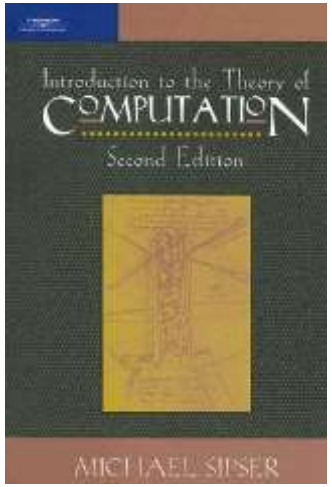
Learning Outcomes

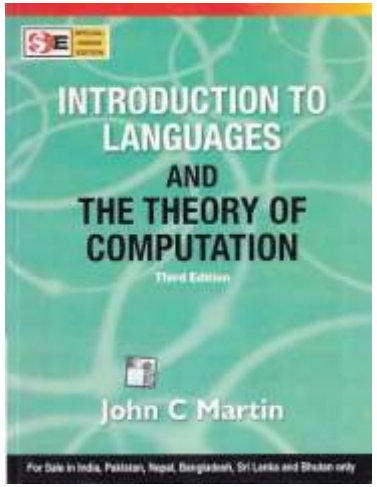
- At successful completion of the course, we will have the knowledge and skills to:
  - Demonstrate advanced knowledge of formal computation and its relationship to languages
  - Distinguish different computing languages and classify their respective types
  - Recognize and comprehend formal reasoning about languages
  - Show a competent understanding of the basic concepts of complexity theory

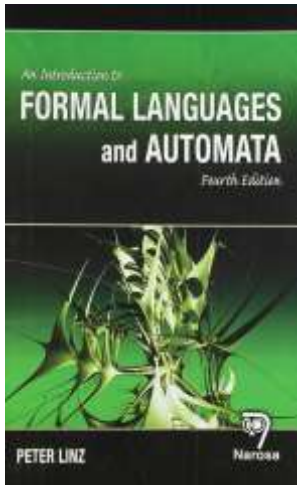
Text and Reference Books

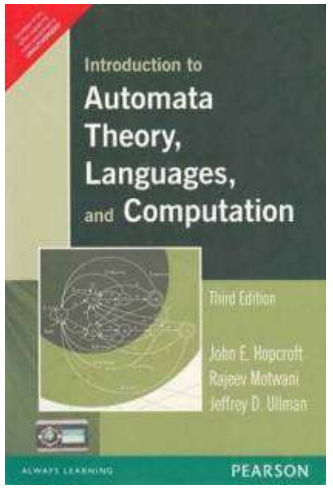
Theory of Computation

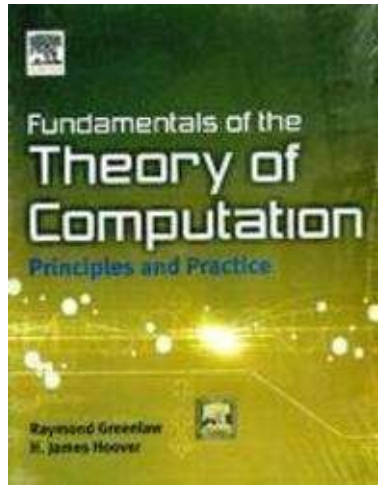












Book Title	Author (s)	Publisher
Theory of Computation	Rajesh K. Shukla	Cengage Learning Pvt. Ltd.
Introduction to the Theory of Computation	Michael Sipser	Thomson Course Technology
Introduction to Languages and The Theory of Computation	John C Martin	The Tata McGraw Hill
An Introduction to Formal Languages and Automata	Peter Linz	Narosa Publishing House
Introduction to Automata Theory, Languages, and Computation	John E. Hopcroft | Jeffery D. Ullman	Narosa Publishing House
Fundamentals of the Theory of Computation	Raymond Green | H. James Hoover	Morgan Kaufmann Publishers

?

THE END