The Islamia University of Bahawalpur

Department of Computer Science & IT

MS(Computer Science)

Course: ***MS101 – Theory of Computation***

Instructor: Dr. Imran Sarwar Bajwa
Contact: imran.sarwar@iub.edu.pk
Teacher Assistant: TBA
Class Timings: 2:00 PM – 05:00 PM
Class Day: Friday
Office Hours: TBA
Class Duration: *48 Credit Hours*

Overview:

Theory of Computation (TOC) is the study of the inherent capabilities and limitations of computers: not just the computers of today, but any computers that could ever be built. By its nature, the subject is close to mathematics, with progress made by conjectures, theorems, and proofs. What sets TOC apart, however, is its goal of 'understanding computation' -- not only as a tool but as a fundamental phenomenon in its own right. This course discusses different issues how computation can happen. Students will study sophisticated methods discovered by the computer scientists over the past decades for managing computer resources, enabling communication, translating programs, designing chips and databases, creating computers and programs that are faster, cheaper, easier to use and more secure as well. In depth, coverage will be made regarding "Computability" and "NP-Completeness". We shall also study various research themes coming under the umbrella of computation theory.

Objectives of the Course:

This course comprises a mathematically sound introduction to the classical and contemporary theory of computation and provides deep insights into the fundamental paradigms of computer Science. Following are major objectives of the course.

- The concept of Algorithm and its analysis are to be introduced and pursued trough out the course.
- Automata will be studied in the context of their applications and in conjunction with the algorithmic problems they pose. State minimization, string matching, LL(1), and bottom-up parsing will be discussed in extensive details.
- Chomsky Normal Form (CNF) and resulting dynamic programming algorithm will be presented with the context-free languages.
- The Turing machine notation will be introduced formally. Simulations between machine models are analyzed quantitatively. A model of random access Turing machines similar to RAMs, will be introduced as well.

- General grammars, μ-recursive functions, and some recursion theory will be succinctly introduced.

- Complexity starts with a proof that there are intractable problems solvable in exponential time. Easy and hard combinatorial problems, including variants of satisfiability, will be introduced, analyzed and their apparent complexities will be contrasted. The intuitions of languages and computational problems will be emerged smoothly.

- A completely new and pedagogically appealing suit of NP-completeness reductions will be elaborated to encompass the state minimization problem of non-deterministic finite automata.

- Logic will be covered using propositional logic in relation to NP-completeness.

Teaching Methodology:

The lectures will be conducted in a discussion environment. The students will be encouraged to participate and ask questions during each class session. Topics from the course outline will be taught in a session and a research paper related to each topic will be given to the students to read it. The research paper will be discussed in the next session. Sessions for group discussion will be arranged to discuss the research papers.

Students will also write one research paper during the semester and submit it in an international conference. A list of international conferences is also available at the end of the course outline. Students are informed to carefully note the submission dates to avoid any inconvenience.

Problems and examples related to each topic will also be done in the class and some exercises will be given to the students as homework.

Home Work:

Some of the exercises will be routine, but others will be more challenging. I do not expect you to solve all of the homework problems, but I hope that you will benefit from working on the some difficult ones. A few hints on the homework assignments:

- **Start early:** Difficult problems are not typically solved in one sitting. Start early and let the ideas come to you the course of a few days.

- **Be rigorous:** Each problem has (sometimes unwritten) requirements that you need to solve it. You may need to study some extra material. Do not restrict to class lectures. Consult other book from library and web.

- **Be Concise:** Express your answers at the proper level of detail. Give enough details to clearly present your solution.

- **Work with Others:** Some of the problems will be difficult, and it will often be helpful to discuss them with your other class-mates. Feel free to form study-groups. Students are encouraged to discuss the difficulties and possible solutions with each other and then each of them should prepare their own solution.

Exams:

Mid term can be closed book/open book (closed discussion). 100% exam will be based on the theory (text and research papers) and exercises discussed in the class.

The final exam will be closed book. 50% exam will be based on the research papers produced by the students and rest of the 50% will be based on the theory (text and research papers) and exercises discussed in the class.

Their will be occasional group discussion sessions, in which students have to discuss the research papers assigned for (home work) study.

Honor Code:

All work submitted for credit must be your own.

Students are supposed to produce original material in their home works, assignments and especially research paper. Plagiarism[1] in any case is not acceptable.

*The student found submitted plagiarized work will be given **zero credit** and may also have financial penalty as well.*

Grading:

- *10% Group Discussions*
- *30% Mid Term Exam*
- *40% Final Term Exam*
- *20% Research Paper*

Pre-requisites:

- Basics of Computer Science
- Discrete Mathematics
- Algorithm Analysis
- Automata Theory

Text and References:

In addition to articles/research papers from various journals and periodicals, material from the following sources will be used in the course.

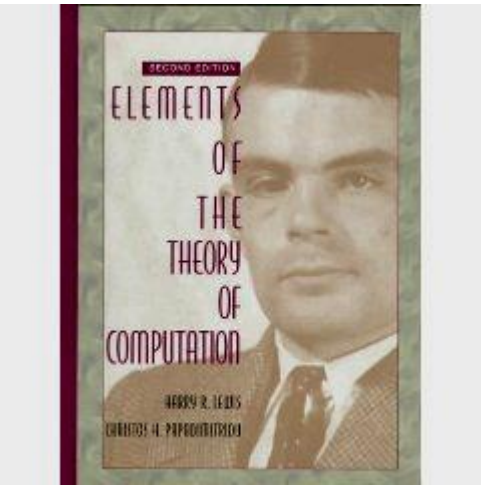
[1] Plagiarism is an act of fraud and it happens if somebody steals and passes of ideas/work of others or uses work of others without their prior permission.

Text Book:

"Elements of The Theory of Computation"

Harry R. Lewis

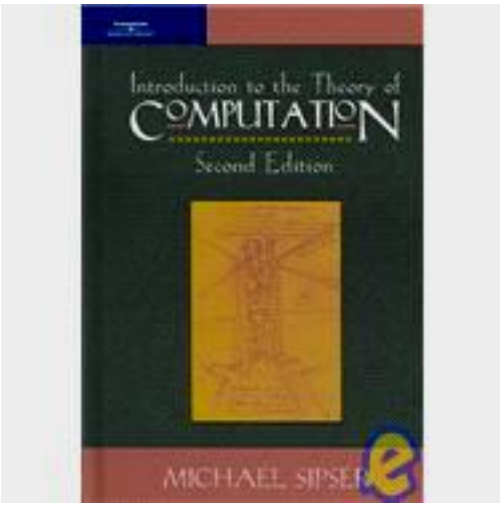
Christos H. P.



Reference Books

"Introduction to the Theory of Computation"

Professor Michael Sipser



Course Contents			
S. No.	***Modules***	***Topics****	***Session***
1.		– Introduction of Course • Class Introduction (*Setting up the field*) • Discussion on Course Outline • How I teach Theory of Computation – How to study research papers? – Glossary of Theory of Computation	Session 1
2.		– Review of Computation Concepts • Sets • Relations and Functions • Closures and Algorithms • Alphabets and Languages – Research Paper & Discussion	Session 2 Session 3 Session 4 Session 5
3.		– Finite Automata • Deterministic • Non-Deterministic • Regular Expressions – Research Paper & Discussion	Session 6 Session 7 Session 8
4.		– Context Free Languages • Context-free Grammars • Parse Trees • Pushdown Automata – Algorithms for context-free Grammars – Non context-free Languages – Research Paper & Discussion	Session 9 Session 10 Session 11 Session 12
5.		– Chomsky Hierarchy of Languages: Regular, Context Free, Context Sensitive and Unrestricted phase structure languages – Research Paper & Discussion	Session 13 Session 14 Session 15
6.		**MID TERM**	Session 16 & 17
7.		– Turing Machine • Computing with Turing Machine • Extension in Turing Machine • Non-Deterministic Turing Machine – Grammars – Research Paper & Discussion	Session 18 Session 19 Session 20

8.		– Reducibility and Decidability – Undecidability • Halting Problem • Universal Problems Turing Machine • Tiling Problem	Session 21 Session 22 Session 23
9.		– Time complexity – Computational Complexity • Class P, NP • Class NP Complete • Class NP Hard – Research Paper & Discussion	Session 24 Session 25 Session 26
10.		– NP –Completeness • Cook's Theorem • NP-Complete Problems • Solving NP-Completeness – Intractability	Session 27 Session 28
11.		– Computability • primitive functions • μ–recursive functions – Approximability	Session 29 Session 30
12.		***Research Paper Defense/Case Study Presentation***	Session 31 Session 32
13.		**Final Exam**	

** Topics can be revisited depending upon the class background and interest.*

Dr. Imran Sarwar Bajwa

Chairman & Associate Professor
Computer Science Department
The Islamia University of Bahawalpur

