CSE 2315 - *Discrete Structures*

Lecture 1- Fall 2019

1 Course Details (see also Syllabus)

- Prerequisites:
  - This course covers many of the theoretical foundations of computer science and provides the basis for many of the later courses (including CSE 3315).
  - To be able to successfully take this course it is important have the basic prerequisites in programming techniques (CSE 1320 or equivalent) and to have passes *Calculus I* and *ENGR 1250*.
- Course Materials:
  - Judith L. Gersting, *Mathematical Structures for Computer Science*, 7th edition, W.H. Freeman and Company, 2014
    If you are using an older edition, you are responsible to identify any differences between editions.
  - Additional course materials will be available electronically. If you have problems obtaining these materials, printed versions will also be available from the instructor.
  - Course announcements will be distributed on Canvas and using e-mail.
  - Changes in assignments or course details, if required, will be posted on Canvas distributed by e-mail. Make sure to read your e-mail frequently.
  - Instructor's e-mail : huber@cse.uta.edu
    Include the course number in the subject line for course-related email.
- Office Hours:
  - Office hours will be given by the instructor in ERB 128 or ERB 522 and are tentatively scheduled M 11:00 - 12:00, M 7:00pm-8:00pm, and W 3:00 - 3:50 and will be held starting Wednesday, August 21.
  - The GTA for the course is Shirin Shirvani (shirin.shirvani@mavs.uta.edu) and her office hours are tentatively scheduled for TTh 11:30am-1:00pm in ERB 505 (or ERB 128)
  - These times are subject to change and will be posted. If for some reason you can not make it to any of these office hours, please inform the instructor.

- Course Schedule:
  - The course is divided into 3 parts of approximately equal size, each followed by an exam.
- Workload and Grading:
  - There are 6 homework assignments (2 per part of the course) accounting for 30 % of the final grade. Assignments are due in or before class on the date indicated on the assignment.
  - The 3 exams in this course are closed book, closed notes. Exams 1 and 2 each cover the content of approximately 2 chapters and count with 20 % each towards the final grade. The final exam is cumulative with a main focus on the last chapters and accounts for 30 % of the grade.
  - You are responsible for all material covered in class, in the assigned sections in the book and in the homework.
  - You are expected to have read the assigned sections in the textbook before each class period.
- Honesty:
  - No collaboration on tests.
  - Solutions to homework assignments are yours only. You may consult with other students but you are not allowed to copy or paraphrase. Solutions have to be original work.
  - Academic dishonest will not be tolerated and violations will result in severe penalties for all parties involved.
  - You are expected to know the University's definitions and policies regarding academic dishonesty.
- To Do:
  - If you have not done so, you have to obtain a copy of the textbook.
  - If are using an older edition of the book, you are responsible for being aware of the differences between the editions.
  - If you can not make any of the posted office hours, let the instructor know so we can potentially move some of the office hours.
  - Make sure you read your UTA email regularly. All announcements will be sent only to your UTA emal.

2 Overview of Content

- This course covers some of the most important theoretical foundations of computer science and automated reasoning.
- It teaches the mathematical method, where the properties of objects depend only on their form and not on some associated semantics as in natural languages. In this course you will learn how to convert problems into these more formal representations, enabling you to represent them in a computer and thus to automate their solution.
- A second objective of the course is to familiarize you with proof techniques which are useful to verify solutions to a large range of problems. These techniques are also used within computer science to test e.g. the correctness of computer programs.
- The course also introduces some of the important mathematical objects such as sets, graphs, and trees which are important for the design and evaluation of algorithms. Use of such, more graphic representations also often simplifies the analysis of a given problem.

Part I: Formal Logic and Mathematical Proof Techniques

- This part of the lecture introduces propositional and predicate logic which allow to formalize mathematical statements in a symbolic form and thus make them accessible to automated reasoning. This is important since natural language (e.g. English) is often ambiguous and can thus not be understood by a computer program.
- Using such formal, symbolic characterizations mathematical proof techniques greatly simplify the task of verifying facts and conclusions. Moreover in some situations fully automatic ways for such verification can be used.

Part II: Sets, Functions, and Matrices

- This part of the course introduces some important concepts in mathematics which are important to formalize more complex problems and which provide mechanisms to analyze them.
- Many of these concepts and techniques are important in all fields of computer science ranging from algorithm design to cryptography and computability.

Part III: Graphs, Trees, Boolean Algebra, and Logic Circuits

- Graphs and trees often provide an efficient means of structuring a given problem. The possibility to represent them in a graphic form makes them easily accessible to us for a large number of problems and the underlying formal structure allows their use in a computer.
- This part of the course introduces the terms associated with graphs and trees and also presents a number of algorithms which solve important graph problems such as reachability or minimum path.
- Boolean algebra and logic circuits, finally, provide a means of designing and analyzing digital circuits or to solve various control problems before actually building the corresponding control mechanism. As such these methods can be important for a wide range of applications in engineering and science.

