# CSC 341: Automata, Formal Languages, and Complexity Theory

Fahmida Hamid

August 31, 2018

## 1 Introduction

### 1.1 Welcome Speech!

Greet the students! Introduce myself, spend 5 mins talking about myself. Tell them about your office hours and how to reach you or schedule an appointment. Then introduce the mentor (David) and the grader (Faizaan). Now talk a little bit about the course policies. Highlight the exams dates, talk about the homework and submission policy, the p-web, collaboration policy, etc.

- Exams
    1. Hour-Exam 01 (October 05 Friday)
    2. Hour-Exam 02 (November 05 Monday)
    3. Final Exam (December 20 Thursday)
- Homework: 8 individual homework, 1 group class presentation

Now start talking about the course: **This is a traditional theory course in CS!** So, what do we cover in this course? Why is it important? Let me start with some examples.

### 1.2 Reduce one problem to another

We try to solve a lot of problems every day: we define algorithms; based on them we write programs; we solve mathematical equations to establish the logical correctness of our statements or hypothesis, and so on. Sometimes we succeed, sometimes we fail. Sometimes we use the same technique to solve two apparently different problems. I am sure you have experienced these situations. That said, I would like to say that **there are a lot of subtle connections between problems.**

Let's say that we have only 2 problems ($A$ and $B$). One has solved problem $B$. No one has been able to solve problem $A$. One day you have discovered that you can model problem $A$ as problem $B$. Great! Now you know the solution to problem $B$ as well as $A$. Once you get the solution of $B$, you just have to convert it back to the parameters of $A$. Here are the steps:

1. Model $A$ as $B$
2. Solve $B$ and produce result $R$
3. Convert $R$ using the parameters of $A$

### 1.2.1 Illustration

**Example:** I get an email from the department chair that we have a meeting at 9:00 AM. It is 8:45 AM. I How quickly can I go to my office? (provided, there are multiple possible routes and I am new in the town). (AKA) finding shortest path

**Example:** I am asked to schedule the final exams so that students do not have time conflicts! (AKA) finding maximum independent sets. Literally, give a small problem.
**Spend some time on the open discussion!**

**Point: One just has to know how to convert one problem to another!**

## 1.3 Problems: Level of Complexity/Difficulty

Let us look back at the two problems we just discussed. Are they equally difficult? Which one looks more difficult? Why do you think that problem $XYZ$ is more difficult than problem $ABC$? We should also know what we mean by complexity.
**Spend some time on the open discussion!**

## 1.4 Proofs: why writing proofs are important?

Once you develop a solution and it looks like that it is working for a set of examples that you have right now, are you confident that it will always succeed?

If you know the solution to a problem, how do you explain it to others? Will they take your words for granted? How will you convince them that it works under all circumstances that may arise? How do you convince others that your solution will work for this problem of size $n = 1, 2, 3, \ldots, n \in N$? Is your approach the best or most efficient one? Can you improve it further?

## 1.5 Intractable and Unsolvable Problems

With the advent of science, have we discovered solutions to all the problems? Can we think of a problem that we don't know how to solve? Or whether there is a universal solution or not? Sometimes the problem sounds very simple but we don't know how to solve it efficiently.

**Example: talk about circuit satisfiability**

## 2 Course Overview

In this course, we focus on three traditionally central area of the theory of computation: automata, computability, and complexity.

## 3 Goals and Objectives

There is no chronological order in which we will attain our goals but here is what we think we are going to practice in the course.

1. to understand a problem and it's corner cases.

2. to learn and practice how to write formal proofs.

3. to understand when a problem is intractable or undecidable

### 3.1 Proof Techniques

Give the hand-out (A proof written in 6 possible ways) to the students. Tell them that we will go over it in the next class.