CS5384: Logic for Computer Scientists

Outline

- History of Logic
- Introduction
- Goals of this course
- Syllabus

History

- Logic is studied in philosophy, math and computer science
- Philosophy: before 1920
  - Studied (principles of) valid reasoning / law of thoughts
- Math: 1920 -1970
  - Logic's major subfields: recursion theory, set theory, model theory and proof theory are major math branches

- Computer Science 1970 –
  - Foundation of computer science
    - Turing machine
    - NP-Completeness
    - …
  - Artificial intelligence: building intelligent agent using logic based approach
  - Programming languages
  - Concurrent, distributed and real-time computing
  - VLSI Design (hardware including CPU etc.)

Introduction

- Motivation: to study valid reasoning
  - we need a *syntax* of formal language to represent what to be reasoned with, or a statement,
  - we need the *semantics* of the language to define the meaning of statements in that language.

- We need to know the difference between syntax and semantics
- Consider a simple language
  - Syntax: the language allows numbers and a symbol $. Example statements in this language include
    - $500
    - $600
  - Semantics of this language
    - One possible meaning: $ -> USD
    - Another possible meaning: $ -> SGD

- Example of valid reasoning
  - Assume we know
    - For any two persons X and Y, X is a parent of Y if X is the father of Y or X is the mother of Y.
    - Pether is the father of John.
  - Is Peter a parent of John?
- Every statement is written in English. It seems that we have some common understanding of *valid reasoning* in the example above.

- In this course we will cover two languages (both syntax and semantics)
  - Propositional logic
  - Predicate logic
- As for valid reasoning, a key component of this course is to study *procedures* to "automate valid reasoning". E.g., procedures will be able to answer the parent question correctly.

A pattern to present a topic

- Start from motivating examples: about ideas behind concepts and/or problems. Set up a clear goal.
- Formal treatment: definitions and proofs
  - Rigorous reading/writing
  - Proof
  - Discussion across the activities above.

Goals of this course

- Learn and practice Logic
  - Develop knowledge about Logic in Computer Science context
  - Be precise in reading/writing/thinking
    - Definitions
    - Proofs
- Skills
  - reading/writing definitions
  - proving statements (reasoning)
  - These are general skills lying at the core of "rigorousness" in almost every discipline particularly STEM. They can help improve our problem solving capacity in other fields.

Expected work

- A lot of preparations are needed for this class:
  - study before class and practice during class.
- Participation during class: asking/answering questions
- Homework, project and exams