Welcome to CS154

Why Study Automata?

What the Course is About

Administrivia

Why Study Automata?

- A survey of Stanford grads 5 years out asked which of their courses did they use in their job.
- Basics like CS106 took the top spots, of course.
- But among optional courses, CS154 stood remarkably high.
  - 3X the score for AI, for example.

How Could That Be?

- Regular expressions are used in many systems.
  - E.g., UNIX a.*b.
  - E.g., DTD's describe XML tags with a RE format like person (name, addr, child*).
- Finite automata model protocols, electronic circuits.
  - Theory is used in *model-checking*.

How? – (2)

- Context-free grammars are used to describe the syntax of essentially every programming language.
  - Not to forget their important role in describing natural languages.
- And DTD's taken as a whole, are really CFG's.

How? – (3)

- When developing solutions to real problems, we often confront the limitations of what software can do.
  - *Undecidable* things – no program whatever can do it.
  - *Intractable* things – there are programs, but no fast programs.
- CS154 gives you the tools.

Other Good Stuff in CS154

- We'll learn how to deal formally with discrete systems.
  - Proofs: You never really prove a program correct, but you need to be thinking of why a tricky technique really works.
- We'll gain experience with abstract models and constructions.
  - Models layered software architectures.

Course Outline

- Regular Languages and their descriptors:
  - Finite automata, nondeterministic finite automata, regular expressions.
  - Algorithms to decide questions about regular languages, e.g., is it empty?
  - Closure properties of regular languages.

Course Outline – (2)

- Context-free languages and their descriptors:
  - Context-free grammars, pushdown automata.
  - Decision and closure properties.

Course Outline – (3)

- Recursive and recursively enumerable languages.
  - Turing machines, decidability of problems.
  - The limit of what can be computed.
- Intractable problems.
  - Problems that (appear to) require exponential time.
  - NP-completeness and beyond.

CS154N

- If you are taking CS154N, you should start coming to class when we enter the Turing-machine material.

Meet the TA's

- Shrey Gupta
- Rohan Jain
- Jia Li

Course Requirements

- Two kinds of homework:
  1. Gradiance homework (automated, straightforward, 20%).
  2. Challenge problems (conventional written work, harder, 20%).
- Two exams:
  - Midterm (20%).
  - Final (Monday June 7, 7-10PM, 40%).

Text

- Hopcroft, Motwani, Ullman, *Automata Theory, Languages, and Computation* 3rd Edition.
- Course covers essentially the entire book.

Gradiance Registration

- The “class token” for this edition of CS154 is **1DC79FE7**.
- Register it at www.gradiance.com/pearson
- Texts come with free access cards.
- Go to www.aw-bc.com/gradiance to purchase or register (link at upper left).

Comments About Gradiance Homework

- The intent is that everyone will get 100% on all homeworks.
- You are allowed to try as many times as you like.
  - Only the last try counts.
- Don’t be afraid to guess and try again.
- You’ll get some advice if you make a mistake.