CS 475: Formal Models of Computation

Mahesh Viswanathan
vmahesh@illinois.edu
3232 Siebel Center

University of Illinois, Urbana-Champaign

Fall 2017
Instructional Staff

- **Instructor:** Mahesh Viswanathan (vmahesh)
- **Teaching Assistant:** Robert Andrews (rgandre2)
- **Office Hours:** See course website. Also by appointment.
- **Contacting Staff:** Use “private note” in Piazza.
Electronic Bulletin Boards

- **Webpage**: General information, course policies, lecture notes
  courses.engr.illinois.edu/cs475/fa2017

- **Piazza**: Announcements, online questions and discussion, contacting course staff. Sign up at
  piazza.com/illinois/fall2017/cs475.

- **Moodle**: Everything related to quizzes, grades, and announcements
  https://learn.illinois.edu/course/view.php?id=24671

- **Gradescope**: Homework submissions and grading
  gradescope.com
Resources for class material

- **Prerequisites:** All material in CS 173, and CS 374
- **Textbook:** Theory of Computation by Dexter Kozen. (Secondary) Automata and Computability by Dexter Kozen. Both available online through the university library.
- **Lecture Notes:** Available on course web-page
- **Video Recording of Lectures:** See course website for link.
Grading Policy: Overview

Total Grade and Weight

- **Homeworks:** 20%
- **Quizzes:** 10%
- **Midterms:** 40% (2 × 20)
- **Finals:** 30%
Homeworks

- One homework every two weeks: Due on Thursday at midnight on Gradescope. Assigned two weeks in advance on Thursday.
- **No late homeworks.** Lowest homework score will be dropped.
- Homeworks may be solved in groups of size at most 3 and each group submits one written solution on Moodle.
- Homework schedule on course webpage.
Homeworks

- One homework every two weeks: Due on Thursday at midnight on Gradescope. Assigned two weeks in advance on Thursday.
- **No late homeworks.** Lowest homework score will be dropped.
- Homeworks may be solved in groups of size at most 3 and each group submits one written solution on Moodle.
- Homework schedule on course webpage.
- Read Homework Guidelines and Academic integrity policies on course website.
Quizzes

- Once every week on Moodle, except the week before exams and the Thanksgiving break.
- Released on Thursday night, and will be due the following Monday at midnight.
- Quizzes aren’t timed. Multiple attempts allowed, with the last attempt being graded.
- There are about 11 quizzes in total. We will drop 2 quizzes.
- Quiz schedule on course webpage.
Examinations

- First Midterm: Tuesday October 3, 7pm to 9pm
- Second Midterm: Wednesday November 1, 7pm to 9pm
- Final Exam: Monday December 18, 8am to 11am
Examinations

- First Midterm: Tuesday October 3, 7pm to 9pm
- Second Midterm: Wednesday November 1, 7pm to 9pm
- Final Exam: Monday December 18, 8am to 11am
- No conflict exam offered unless you have valid excuse.
Examinations

- First Midterm: Tuesday October 3, 7pm to 9pm
- Second Midterm: Wednesday November 1, 7pm to 9pm
- Final Exam: Monday December 18, 8am to 11am
- No conflict exam offered unless you have valid excuse.
- Midterms will only test material since the previous exam
- Final Exam will test all the course material
Part I

Course Overview
Overview

Computational Complexity
Overview

Computational Complexity

of decision problems

Viswanathan CS 475
Overview

Computational Complexity

of decision problems

in models of computations

Nondeterminism
Randomness
Parallel
Interaction
Time
Space
Parallelism
Randomness
Advice
Overview

Computational Complexity

- of decision problems
- in models of computations
- w.r.t. computational resources
Overview

Computational Complexity

of decision problems
in models of computations
w.r.t. computational resources

Nondeterminism
Overview

Computational Complexity

- of decision problems
- in models of computations
- w.r.t. computational resources

Nondeterminism
- Randomness
Overview

Computational Complexity

- of decision problems
- in models of computations

w.r.t. computational resources

Nondeterminism
Randomness
Parallel
Overview

Computational Complexity

- of decision problems
- in models of computations

w.r.t. computational resources

- Nondeterminism
- Randomness
- Parallel
- Interaction
Overview

Computational Complexity

- of decision problems
  - in models of computations
- w.r.t. computational resources

Nondeterminism
- Randomness
- Parallel
- Interaction
- Time
Overview

Computational Complexity

- of decision problems
- in models of computations
- w.r.t. computational resources

- Nondeterminism
- Randomness
- Parallel
- Interaction
- Time
- Space
Overview

Computational Complexity

- of decision problems
- in models of computations
- w.r.t. computational resources

- Nondeterminism
- Randomness
- Parallel
- Interaction
- Time
- Space
- Parallelism
Overview

Computational Complexity

of decision problems in models of computations w.r.t. computational resources

Nondeterminism
Randomness
Parallel
Interaction
Time
Space
Parallelism
Randomness
Overview

Computational Complexity

- of decision problems
- in models of computations
- w.r.t. computational resources

- Nondeterminism
- Randomness
- Parallel
- Interaction
- Time
- Space
- Parallelism
- Randomness
- Advice
High-Level Questions

What are the limits of computation?
High-Level Questions

- What are the limits of computation?
- Is finding proofs as easy as checking their correctness?
High-Level Questions

- What are the limits of computation?
- Is finding proofs as easy as checking their correctness?
- Is every efficient sequential algorithm parallelizable?
High-Level Questions

- What are the limits of computation?
- Is finding proofs as easy as checking their correctness?
- Is every efficient sequential algorithm parallelizable?
- Can every (time) efficient algorithm be converted into one that uses a small amount of space?
High-Level Questions

- What are the limits of computation?
- Is finding proofs as easy as checking their correctness?
- Is every efficient sequential algorithm parallelizable?
- Can every (time) efficient algorithm be converted into one that uses a small amount of space?
- Can every efficient randomized algorithm be converted into an (efficient) deterministic one?