Some key ideas, techniques, tools and applications

- **Random selection**
  - choose “typical” element of a set, avoiding rare “bad” elements. Example: min-cut
- **Random ordering and backwards analysis**
  - Order input randomly, express probability in terms of “output”structure created. Examples: computational geometry, data structures.
- **Random sampling as algorithmic tool**
  - Sample to get representative subproblem which can be solved efficiently. Examples: median-finding, MST
- **Fingerprinting and hashing**
  - FPs are short signatures for long string; hashing is technique for storing set and implementing dictionary operations

Some key ideas, techniques, tools and applications

- **Fingerprinting and hashing**
  - FPs are short signatures for long string; hashing is technique for storing set and implementing dictionary operations
- **Use of limited independence**
  - Enables reduction in amount of randomness needed. Examples: universal hashing and perfect hashing.
- **Abundance of witnesses**
  - Want to determine if input has a certain property (e.g. is x prime?). Find a "witness" to fact. If search space large, but witnesses abundant, can search randomly.
- **Minimax theorem**
  - Randomized complexity ⬄ average case analysis
- **Probabilistic method**
  - Use probabilistic argument to prove non-probabilistic mathematical statements.

Some key ideas, techniques, tools and applications

- **Tail bounds**
  - Crucial for bounding deviation from expectation
- **Load balancing**
  - Balls in bins type problems. Example: allocation of resources in a distributed environment.
- **Randomized rounding**
  - Technique for transforming fraction solutions or vector solutions -> integral solutions. Most important for approximately solving NP-complete problems.
- **Random walks, Markov chains, Markov Chain Monte Carlo**
  - Techniques for estimating probabilities of interesting events, approximately counting interesting objects, sampling
- **Martingales**
  - Collection of tools for reasoning about certain kinds of naturally arising stochastic processes that correspond roughly to fair gambling.

Some key ideas, techniques, tools and applications

- **Dimensionality reduction**
  - E.g. Johnson-Lindenstrauss
- **Random graphs and other random structures**
  - Properties, 0-1 laws, etc.
- **Other techniques**
  - E.g. Lovasz Local Lemma, Chen-Stein method, etc.
- **Entropy and Information Theory**
  - With applications to error-correction codes and compression

Administrivia

- **Anna Karlin**
  - CSE 594, karlin@cs
  - Office hours by appointment

- **Paris Koutris**
  - CSE 378, pkoutris@cs
  - Office hours: TBD

- **Course web page:**
  - http://www.cs.washington.edu/525

Plan for Course

- First half or a bit more:
  - mix of introductory lectures on various of topics just mentioned, with a focus on introducing key techniques, and for each technique, at least one application.
- Second half:
  - Techniques and exciting applications from the last 5 years.
  - Goal: to push us to the frontier

Background expected

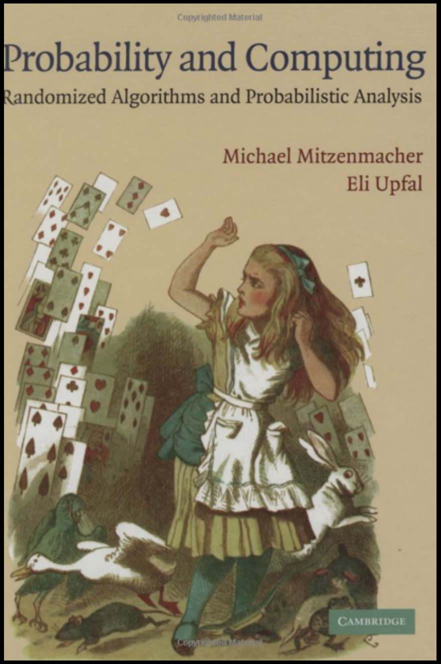
- Introductory probability at the level of CSE 312:
  - Probability space, random variables, basic distributions, independence, conditional probability, expectation, …
- Algorithms <= CSE 412
- “Mathematical maturity”
  - Linear programming
  - Linear algebra

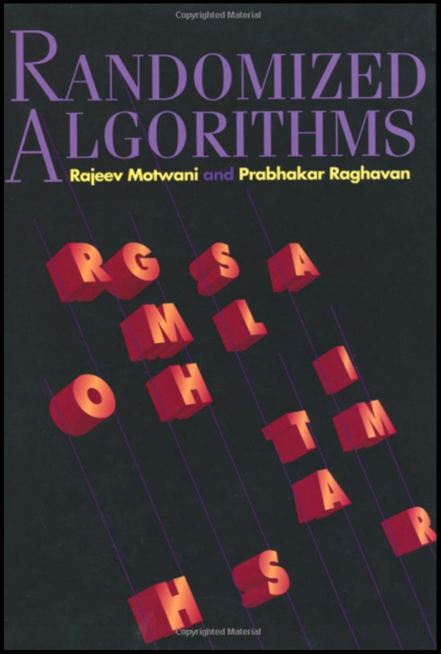
Workload

- Approximately 4 problems sets.
- Paper and presentation on a research paper relevant to the course.
  - Can work in pairs.
  - Paper must be approved by May 1.
  - 30 minute presentation during the last 2 weeks of quarter.
  - After the presentation, I'll ask you to delve into the details of some specific aspect for the final version of the paper.
- Final

Other

- Books


Copyrighted Material
Probability and Computing
Randomized Algorithms and Probabilistic Analysis
Michael Mitzenmacher
Eli Upfal
CAMBRIDGE



- No class April 16!