CS580: Algorithm design and analysis

Lecture 1

Logistics

- Instructor: Elena Grigorescu
- Course website: http://www.cs.purdue.edu/homes/egrigore/580FT13/
- TAs: Leo Osvald, Someone Else

- Discussion website: Piazza.com (CS580)
  Send all your comments via this site.

- Recommended texts:
  Algorithm design, by Kleinberg and Tardos
  Intro to algorithms, by Cormen, Leiserson, Rivest, Stein

Logistics

- Homework policy:

PSets due every 2 weeks, in hard copy

You can collaborate with other students

You must write down your own solutions

We might grade a subsets of the assigned problems (unknown to you)

- Grading:

35% for homework

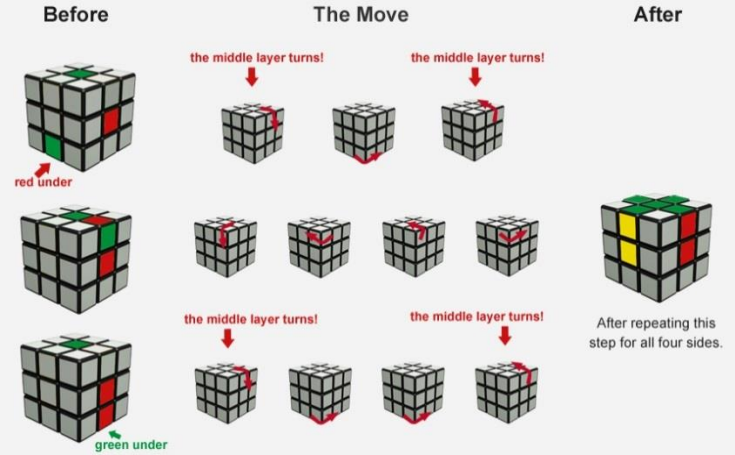
25% for the midterm

35% for the final (no makeup exams)

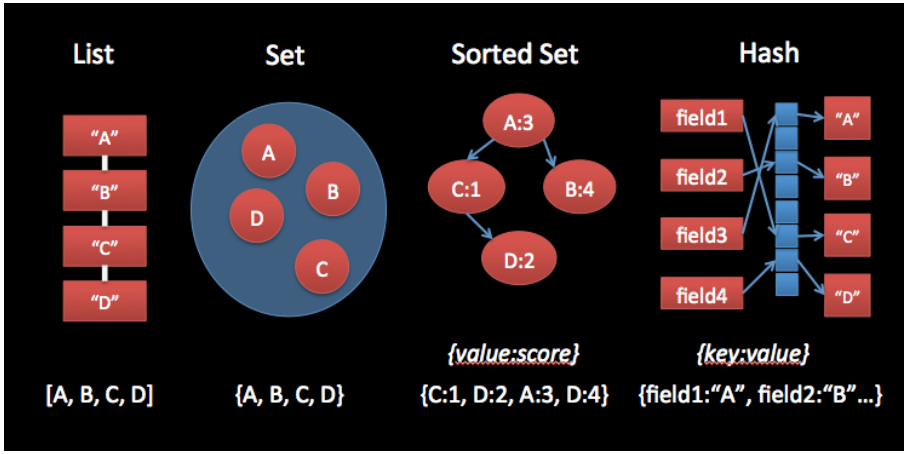
5% for class participation (answer your colleagues' questions on Piazza)

What is this course about?

- Algorithms



- Data structures



What aspects are important?

Computational aspects:

- Efficiency (time and space)
- Feasibility

  Is the problem decidable? is it NP-complete?

Design aspects:

- Robustness
- Scalability
- Modularity
- Simplicity

Prerequisites

- **Familiarity with mathematical proofs (induction, contradiction)**
  We will be proving the correctness of algorithms.

- Asymptotic notation (O(n), o(n), etc.)

- Basic data structure: linked lists, stacks, arrays, trees

- Basic algorithm paradigms: divide and conquer, greedy algorithms, dynamic programming

- Basic algorithms: depth-, breadth- first search, basic sorting algorithms

- Notions of computability and complexity: Turing machines, P, NP

Goals of the course

- Learn how to formulate real-world questions using mathematical abstractions
- Learn about algorithms for classical problems
- Learn how to analyze algorithms formally
- Develop/practice problem-solving, technical-writing, and analytical-thinking skills

Topics

- Scheduling problems: interval scheduling, interval partitioning
- Minimum spanning trees
  applications in network design, approximation algorithms, reducing data storage, cluster analysis
- Data compression: how can we encode text in bits?
- Fast Fourier Transform (FFT): How can we perform matrix/polynomial multiplication fast?
  applications to numerical analysis, digital media (jpg, dvd), engineering

Topics (contd)

- Max flow, min cut
  applications to data mining, image segmentation, security of statistical data
- Linear programming
  applications to max flow, min cut, optimizations
- Complexity: reducibility, intractability, NP completeness
- Approximation algorithms
- Randomized algorithms. Analysis tools.
- Sublinear algorithms

Techniques

- Greedy algorithms
- Divide and conquer
- Dynamic programming
- Probabilistic tools

Remainder of this lecture

(on the board)

The Stable Matching Problem:

algorithm and analysis.