

# Theory of Algorithms VI

Conclusion

Guoqiang Li

School of Software, Shanghai Jiao Tong University

# Algorithm Design

# Algorithm Design

- Basic methodologies:

# Algorithm Design

- Basic methodologies:
  - Algorithms on Lists, Trees and Graphs

# Algorithm Design

- Basic methodologies:
  - Algorithms on Lists, Trees and Graphs
  - Divide and Conquer

# Algorithm Design

- Basic methodologies:
  - Algorithms on Lists, Trees and Graphs
  - Divide and Conquer
    - Master Theorem

# Algorithm Design

- Basic methodologies:
  - Algorithms on Lists, Trees and Graphs
  - Divide and Conquer
    - Master Theorem
  - Recursion
- Advanced topics:

# Algorithm Design

- Basic methodologies:
  - Algorithms on Lists, Trees and Graphs
  - Divide and Conquer
    - Master Theorem
  - Recursion
- Advanced topics:
  - Dynamic Programming



# Algorithm Design

- Basic methodologies:
  - Algorithms on Lists, Trees and Graphs
  - Divide and Conquer
    - Master Theorem
  - Recursion
- Advanced topics:
  - Dynamic Programming
  - Greedy Algorithms

# Algorithm Design

- Basic methodologies:
  - Algorithms on Lists, Trees and Graphs
  - Divide and Conquer
    - Master Theorem
  - Recursion
- Advanced topics:
  - Dynamic Programming
  - Greedy Algorithms
  - Linear Programming

# Algorithm Design

- Basic methodologies:
  - Algorithms on Lists, Trees and Graphs
  - Divide and Conquer
    - Master Theorem
  - Recursion
- Advanced topics:
  - Dynamic Programming
  - Greedy Algorithms
  - Linear Programming
  - Approximation Algorithms

# Algorithm Design

- Basic methodologies:
  - Algorithms on Lists, Trees and Graphs
  - Divide and Conquer
    - Master Theorem
  - Recursion
- Advanced topics:
  - Dynamic Programming
  - Greedy Algorithms
  - Linear Programming
  - Approximation Algorithms
  - Randomized Algorithms
  - Computational Geometry
  - Algorithms on Data
  - ...

# Algorithm Analysis

# Algorithm Analysis

- Big-*O* Notation

# Algorithm Analysis

- Big-*O* Notation
- Advanced Methodology:

# Algorithm Analysis

- Big-*O* Notation
- Advanced Methodology:
  - Probability Analysis



# Algorithm Analysis

- Big-*O* Notation
- Advanced Methodology:
  - Probability Analysis
  - Amortized Analysis

# Algorithm Analysis

- Big-*O* Notation
- Advanced Methodology:
  - Probability Analysis
  - Amortized Analysis
  - Competition Analysis

# Standard Algorithms

# Standard Algorithms

- Sorting

# Standard Algorithms

- Sorting
- Searching & Hashing

# Standard Algorithms

- Sorting
- Searching & Hashing
- Strongly connected components

# Standard Algorithms

- Sorting
- Searching & Hashing
- Strongly connected components
- Finding shortest paths in graphs

# Standard Algorithms

- Sorting
- Searching & Hashing
- Strongly connected components
- Finding shortest paths in graphs
- Minimum spanning trees in graphs



# Standard Algorithms

- Sorting
- Searching & Hashing
- Strongly connected components
- Finding shortest paths in graphs
- Minimum spanning trees in graphs
- Matchings in bipartite graphs

# Standard Algorithms

- Sorting
- Searching & Hashing
- Strongly connected components
- Finding shortest paths in graphs
- Minimum spanning trees in graphs
- Matchings in bipartite graphs
- Maximum flows in networks

# Data Structure

# Data Structure

- Link lists
- Trees, graphs
- Heaps

# Data Structure

- Link lists
- Trees, graphs
- Heaps
- Balance trees, red-and-black trees
- Kripke structure, automata

# Data Structure

- Link lists
- Trees, graphs
- Heaps
- Balance trees, red-and-black trees
- Kripke structure, automata
- Priority queue
- Disjoint set
- ordered binary decision diagrams (OBDD)

# Data Structure

- Link lists
- Trees, graphs
- Heaps
- Balance trees, red-and-black trees
- Kripke structure, automata
- Priority queue
- Disjoint set
- ordered binary decision diagrams (OBDD)
- ...

# Computational Complexity

Church-Turing Thesis

NP Problems



# Computational Complexity

Church-Turing Thesis

NP Problems

Complexity Class

The Door of Algorithms will Open!