

Design and Analysis of Algorithms

Course Information

- 1 Why Study Algorithm?
- 2 How to Study Algorithm?

1 Why Study Algorithm?

2 How to Study Algorithm?

Two ideas changes the world!

Typography

1448, German, Johann Guternberg: print books by putting together movable metallic pieces



- literacy spread \Rightarrow Dark Ages ended \Rightarrow human intellect was liberated \Rightarrow science and technology triumphed \Rightarrow Industrial Revolution happened

imagine a world in which only an elite could read lines

Typography

1448, German, Johann Guternberg: print books by putting together movable metallic pieces



- literacy spread \Rightarrow Dark Ages ended \Rightarrow human intellect was liberated \Rightarrow science and technology triumphed \Rightarrow Industrial Revolution happened

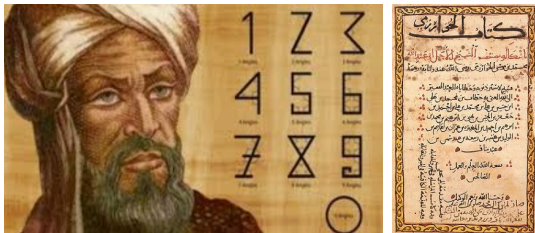
imagine a world in which only an elite could read lines

But others insists that the key development was not typography,
but *algorithm*

Algorithm

Origin: decimal system

- 10 symbols \Rightarrow even large numbers can be expressed compactly (invented in India around AD 600)
- basic methods for add, mul, div, even square roots and π (9th century, Arabic, Baghdad, Al-Khwarizmi)

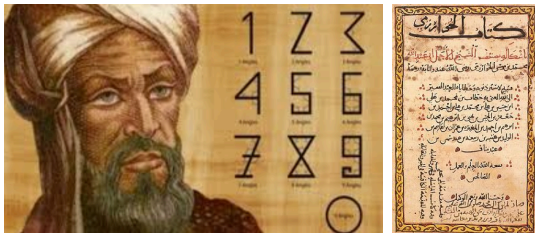


These procedures are precise, unambiguous, mechanical, efficient, correct \leadsto algorithms

Algorithm

Origin: decimal system

- 10 symbols \Rightarrow even large numbers can be expressed compactly (invented in India around AD 600)
- basic methods for add, mul, div, even square roots and π (9th century, Arabic, Baghdad, Al-Khwarizmi)



These procedures are precise, unambiguous, mechanical, efficient, correct \leadsto algorithms

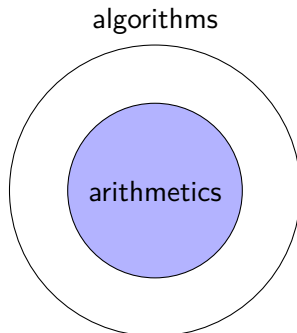
Back to 1448: imagining how to add/mul two Roman numbers:
MCDXLVIII+DCCCXII? fingers are not enough

Algorithm Etymology

Spread to Europe around 12th century → plays an enormous role in Western civilization (science and technology, commerce and industry)

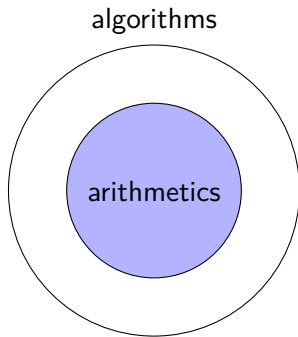
Algorithm Etymology

Spread to Europe around 12th century → plays an enormous role in Western civilization (science and technology, commerce and industry)



Algorithm Etymology

Spread to Europe around 12th century → plays an enormous role in Western civilization (science and technology, commerce and industry)



Computer era: evolve to embody the positional system and arithmetic unit \rightsquigarrow scientists develop algorithms for all kinds of problems — ultimately change the world

Why Study Algorithms

Internet. Web search, packet routing, distributed file sharing, ...

Computer graphics. movies, video games, virtual reality, ...

Multimedia. MP3, JPG, DivX, HDTV ...

Artificial Intelligence. face recognition, PS, more AI algorithms

Social networks. recommendations, news feeds, advertisements, ...

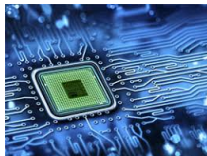
Computers. circuit layout, databases, caching, compilers, ...

Biology. human genome project, protein folding, ...

Physics. N -body simulation, particle collision simulation, ...

Importance: Look around you

Google
YAHOO!
bing



Algorithms interesting and useful.
We live in the algorithm world!

Cryptographic Algorithms

Most algorithms focus on solving problems efficiently

- make us live in a better world

Good man and bad man live in the same world

- good man need *cryptographic algorithms* to protect them from bad man: enjoying the benefits in a secure manner

Cryptographic algorithms ensure there is no *efficient algorithms* against some problems



Fundamental of Computer Sciences

Algorithm design and analysis

- widespread applications
- fundamental and core part of computer science

Fundamental of Computer Sciences

Algorithm design and analysis

- widespread applications
- fundamental and core part of computer science

Turing Awards: (1966-2005) 50 persons win Turing awards

- algorithm design: 10
- computing and complexity theory: 7
- cryptography: ≥ 8

Fundamental of Computer Sciences

Algorithm design and analysis

- widespread applications
- fundamental and core part of computer science

Turing Awards: (1966-2005) 50 persons win Turing awards

- algorithm design: 10
- computing and complexity theory: 7
- cryptography: ≥ 8

$\mathcal{P} \stackrel{?}{=} \mathcal{NP}$ is one of the most important questions in this century

1 Why Study Algorithm?

2 How to Study Algorithm?

Contents of This Course

Preliminary about algorithms

- mathematical background
- data structure

Contents of This Course

Preliminary about algorithms

- mathematical background
- data structure

Design paradigm and analysis methods

- divide-and-conquer
- greedy strategy
- dynamic programming
- backtracking and trimming technique

Contents of This Course

Preliminary about algorithms

- mathematical background
- data structure

Design paradigm and analysis methods

- divide-and-conquer
- greedy strategy
- dynamic programming
- backtracking and trimming technique

Advanced topics

- complexity theory
- randomized algorithms
- approximate algorithms

Goal of this Course

Emphasize **critical thinking**: believe your own reasoning, do not easily repeat or follow

Goal of this Course

Emphasize **critical thinking**: believe your own reasoning, do not easily repeat or follow

Algorithm design: Master **problem-solving** method

- 1 abstract and formalize problem
- 2 solve it efficiently and correctly using algorithms
- 3 prove its correctness

Goal of this Course

Emphasize **critical thinking**: believe your own reasoning, do not easily repeat or follow

Algorithm design: Master **problem-solving** method

- 1 abstract and formalize problem
- 2 solve it efficiently and correctly using algorithms
- 3 prove its correctness

Algorithm analysis: Develop **rigorous analysis** skills

- know how to evaluate the performance of algorithms

Goal of this Course

Emphasize **critical thinking**: believe your own reasoning, do not easily repeat or follow

Algorithm design: Master **problem-solving** method

- 1 abstract and formalize problem
- 2 solve it efficiently and correctly using algorithms
- 3 prove its correctness

Algorithm analysis: Develop **rigorous analysis** skills

- know how to evaluate the performance of algorithms
-

Tips

- theory: think rigorously and keep ask yourself why
- practice: implement algorithms using your favorite programming languages

Course Website

<https://yuchen1024.github.io/teaching/SDU/2021/Algorithms/algorithms.html>

Syllabus

Assignments

- electronic submission
- graded for correctness, clarity, conciseness, rigor, and efficiency
- recommendation: using \LaTeX template for writing solutions
- no collaboration, no Google

Lecture slides

...

总成绩 = $0.2 \times$ 平时成绩 + $0.2 \times$ 课后作业 + $0.6 \times$ 考试成绩

References and Resources

Online resources

- leetcode
- online judging system: ZOJ, POJ

References and Resources

Online resources

- leetcode
- online judging system: ZOJ, POJ

Textbooks

- Algorithms. Sanjoy Dasgupta, Christos Papadimitriou, and Umesh Vazirani. The McGraw-Hill Companies, 2008.
- 算法设计与分析 (第二版). 屈婉玲, 刘田, 张立昂, 王捍贫. 清华大学出版社, 2016.2.



Figure: 屈婉玲