Design and Analysis of Algorithms
Introduction

Si Wu

School of Computer Science and Engineering

E-mail: cswusi@scut.edu.cn

- Part I: About the course
- Part II: About algorithms
  - What are algorithms?
  - Why are they important to study?

Part I: About the course

Course Information

- Lecturer: Si Wu (吴斯)
- E-mail: cswusi@scut.edu.cn
- Office Room： B3-302

If you have any questions, please feel free to contact me by email. Please list your name or student ID when you send me an email….

Course Information

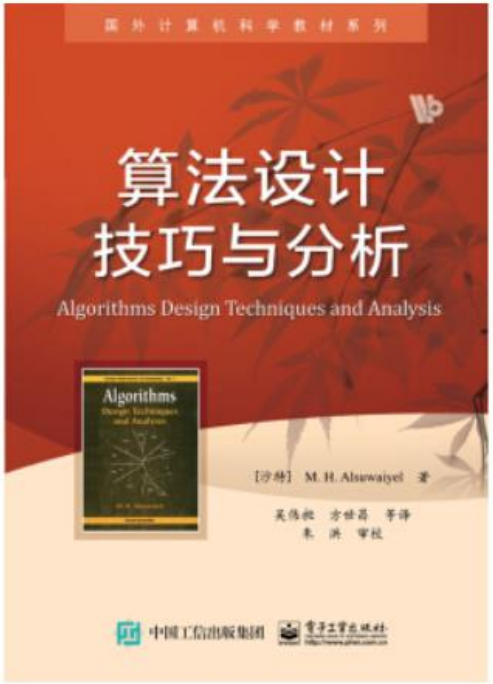
- Teaching Assistants：
  - Wenhao Wu (吴文豪)

    E-mail: 1565865638@qq.com
  - Yi Liu (刘艺)

    E-mail: 1337545838@qq.com

Course Information

- Reference
  - Algorithms Design Techniques and Analysis. (Saudi Arabia) M. H. Alsuwaiyel. Publishing House of Electronic Industry.



Main Topics

- Algorithm Analysis
- Sorting algorithms
- Recurrence
- Divide and Conquer
- Dynamic Programming
- Greedy Algorithms
- Linear Programming
- Network Flow
- Approximation
- P & NP Problems

Course Information

- **Couse Time**： Weeks 1-9 &12-18, 64 lessons
  (including 16 lessons for experiments)
  2pm-4:30pm, Wednesday
- **Final Grade**：

  Performance + Experiments (30%)

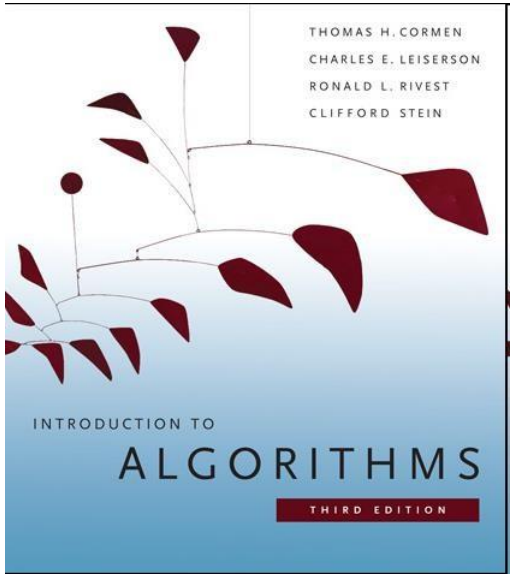
  Final Examination (70%)

Course Information

- Online Judge：
  - http://www.scut.edu.cn/ACM/
    (South China University of Technology)
  - http://acm.zju.edu.cn/onlinejudge/
    (Zhejiang University)
  - http://poj.org (Peking University)

Good reference

- The following book containing many materials that we don't have time to cover.



Introduction to Algorithms, 3rd ed, T.H. Cormen, C.E. Leiserson, R.L. Rivest, C. Stein, *MIT Press*, 2009.

About the flavor of the course

- It's more of a math flavor than a programming one.
- You will need to write pseudo-code, and implement it using C/C++...
- You will design and analyze, think and prove (rather than code)

Prerequisites

- Officially:

  DISCRETE MATHEMATICS

  PRGRAMING

  DATA STRUCTURES

- Effectively: Basic mathematical maturity

  functions, polynomial, exponential;

  proof by induction;

  basic data structure operations (stack, queue, …);

  basic math manipulations…

Experiment Policy

- Discussions and googling on web are allowed in general
- But you have to write down the solution by yourself
- And you should fully understand what you write.

Zero tolerance for cheating/plagiarism

- You may get 0 score for this course
- Will check your codes by software; scores of both the codes provider and the copier will be 0 once the cheating/plagiarism behavior is confirmed

Suggestions

- In class:
  - Try to come on time.
  - Try your best to get more involved in the class.
  - Treat experiments seriously

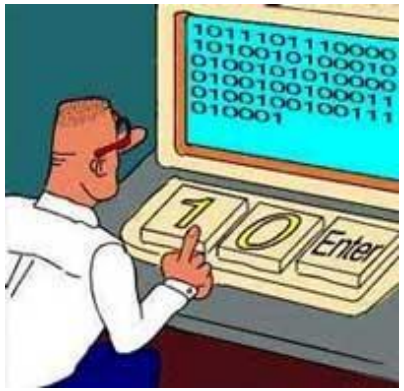
Suggestions

- Your suggestion will be highly appreciated.
  - Please send me an e-mail
- Any questions about the course?
- My questions:
  - What do you like to learn from this course?
  - What excite you the most in general?

Part II: About algorithms

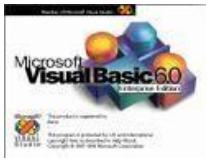
Factors of Programming

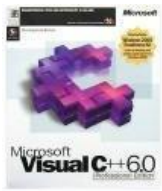
- **Programming Languages?**



BASIC PASCAL C 程序设计语言

C++ JAVA C#





Algorithms

- **Algorithm. (webster.com)**

--A well-defined computational procedure that takes some value, or set of values, as input and produces some value, or set of values, as output.

--Broadly: a step-step procedure for solving a problem or accomplishing some end especially by a computer.

--issues: correctness, efficiency (amount of work done and space used), storage (simplicity, clarity), optimality .etc.

Why study algorithms?

Importance of algorithms

- **Problem: sorting 10,000,000 integers**
  - Case 1: Computer A executes one billion instructions per second(1GHz), an algorithm taking time roughly equal to $2n^2$ to sort n integers.
  - Case 2: Computer B executes one hundred million instructions per second(100MHz), an algorithm taking time roughly equal to 50nlogn to sort n integers.
  - Case 1:

$$\frac{2\times(10^7)^2\,\text{instructions}}{10^9\,\text{instructions}/\text{second}} = 200000\text{seconds} \approx 55\text{hours}$$

  - Case 2:

$$\frac{50\times10^7\times\log10^7\,\text{instructions}}{10^8\,\text{instructions}/\text{second}} = 105\text{seconds}$$

Importance of algorithms

Run time (nanoseconds)		$1.3\ N^3$	$10\ N^2$	$47\ N \log_2 N$	$48\ N$
Time to solve a problem of size	1000	1.3 seconds	10 msec	0.4 msec	0.048 msec
	10,000	22 minutes	1 second	6 msec	0.48 msec
	100,000	15 days	1.7 minutes	78 msec	4.8 msec
	million	41 years	2.8 hours	0.94 seconds	48 msec
	10 million	41 millennia	1.7 weeks	11 seconds	0.48 seconds
Max size problem solved in one	second	920	10,000	1 million	21 million
	minute	3,600	77,000	49 million	1.3 billion
	hour	14,000	600,000	2.4 billion	76 billion
	day	41,000	2.9 million	50 billion	1,800 billion
N multiplied by 10, time multiplied by		1,000	100	10+	10

Information Explosion

- 988EB (1EB = 1024PB) data will be produced in 2010 (IDC) ⇔ 18 million times of all info in books
- IT
  - 850 million photos & 8 million videos every day (Facebook)
  - 50PB web pages, 500PB log (Baidu)
- Public Utilities
  - Health care (medical images - photos)
  - Public traffic (surveillance - videos)
- …

Research Frontier and Hot

- 《Science》: Special Online Collection: Dealing with Data
  - In this, *Science* joins with colleagues from *Science Signaling*, *Science Translational Medicine*, and *Science Careers* to provide a broad look at the issues surrounding the increasingly huge influx of research data. This collection of articles highlights both the challenges posed by the data deluge and the opportunities that can be realized if we can better organize and access the data.

11 FEBRUARY 2011 VOL 331
SCIENCE
www.sciencemag.org

- 《Nature》:

Big data, but are we ready?

Oswaldo Trelles, Pjotr Prins, Marc Snir and Ritsert C. Jansen

What kind of problems

- Human Genome Project
  - 100,000 genes， sequences of the 3 billion chemical base pairs
- Internet
  - Finding good routes on which the data will travel
  - Search engine
- Electronic commerce
  - Public-key cryptography and digital signatures
- Manufacturing
  - Allocate scarce resources in the most beneficial way
- …

About the Course

- **Design and Analysis**
  - How can I propose an algorithm for a specific problem?
  - Is the algorithm good enough?

Analysis of algorithms

•The theoretical study of computer-program performance and resource usage.

- **what's more important than performance?**

--modularity
--correctness
--maintainability
--functionality
--robustness
--user-friendliness
--programmer time
--simplicity
--extensibility
--reliability