

Introduction to Algorithms

算法导论

北京航空航天大学 软件学院
School of Software, Beihang University

宋 友 (Song You)

Email : songyou@buaa.edu.cn

Office: 新主楼C307

Chapter 0

Preface

Algorithms (算法)

教师 宋 友
罗 川

崔绍锜 cuisk@buaa.edu.cn

闫佳豪 21377242@buaa.edu.cn

助教 杨振炜 20377408@buaa.edu.cn

闵家旭 minjiaxu@buaa.edu.cn

云惟彬 3338608544@qq.com

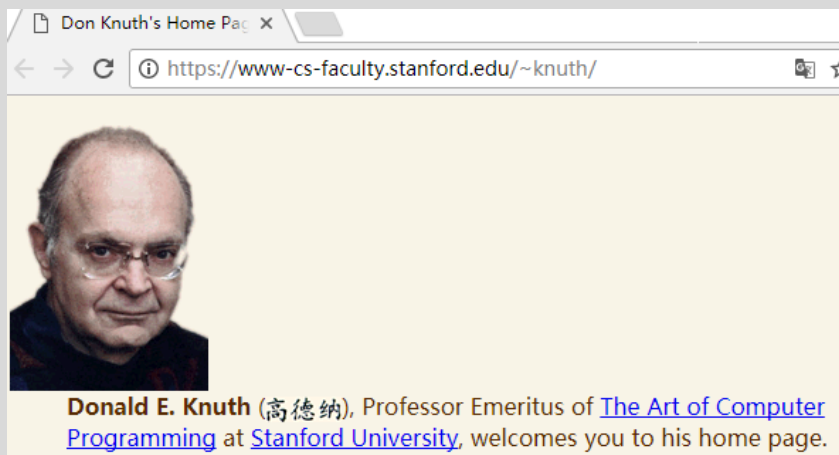
王泓翔 23371110@buaa.edu.cn

序：分析算法的幸福

分析算法的人享有双重的幸福：

- 一方面，他们能够体验到优雅数学模式纯粹的美，这种模式存在于优美的计算过程之中；
- 另一方面，当他们的理论使得其他工作能够做得更快、更经济时，他们能够得到实际的褒奖。

—Donald E. Knuth



[Don Knuth's Home Page \(stanford.edu\)](https://www-cs-faculty.stanford.edu/~knuth/)

唐纳德·克努特（1938-），1974年获得图灵奖，常译为**克鲁斯**，中文名高德纳，算法和程序设计理论与技术的先驱者，经典巨著《**计算机程序设计艺术**》(The Art of Computer Programming)的作者，计算机排版系统TEX和METAFONT的发明者，他因这些成就和大量创造性的影响深远的著作而誉满全球。被誉为“人工智能之父”。

序：技术的演变

最近二十年的热门IT词汇（年份是大概数）

- 2003, 云计算
- 2004, 物联网
-
- 2014, 大数据 (Big Data)
- 2015, 互联网+
- 2016, 虚拟现实 (VR, AR)
- 2017, 人工智能 (AI)
- 2018, 区块链
- 2019, 5G的相关信息科技概念
- 2020, 2021, 元宇宙
- 2022, 2023, OpenAI-ChaGPT
- 2024, OpenAI-Sora

**这些技术的灵魂
都是算法！**



人工智能发展的三大引擎：算法、算力、算料（即数据）。 算法：人工智能的智慧之源。

**年年都有新概念，有的还几起几落，.....，
概念在变、技术在变、语言在变，唯有算法永恒！**

序：算法是财富的核心

《未来简史》书摘：随着算法将人类挤出就业市场，财富和权力可能会集中在拥有强大算法的极少数精英、寡头手中。



观察1：航空公司低价票越来越少，航班满座率越来越高，几个航班通常合并，很多航班的误点时间精准控制在2小时以内，……（航空公司就是寡头之一）（2020~2022年的航班少是个例外，但这期间的国际机票却是几万一张，还一票难求）

观察2：你看来一段小视频，天天给推类似的视频，…

观察3：你投入积蓄买股票，想要赚点零花钱，另一端却是强大的量化交易对手，…

观察4：你投入大量时间、情感、金钱，聊的很开心的网友，可能是一个算法，…

序：算法是软件的灵魂（其实也是硬件的灵魂）

- 计算机硬件和软件可以看成两个相互依存但又对立的两个不同体。
- 算法是软件的核心，也是硬件的灵魂（如：硬件是超大规模集成电路，电信号（0，1）能用来表示逻辑运算，但能进一步进行关系运算、推理、算术运算等，这本身就是算法，或者说，硬件灵魂也是算法【运算规则是算法】）
- 软件：狭义地讲，就是程序，
算法 + 数据结构 = 程序
- 本课程主要从软件视角来学习算法。
- 算法这么重要，自然地，也是很难学的！

有一个人因为一句话而得到了图灵奖：Nicklaus Wirth（1984年，图灵奖）——Pascal之父，这个公式对计算机领域的影响程度足以类似物理学中爱因斯坦的“ $E=MC^2$ ”（卡脖子现状的本质就是缺芯少魂，魂，算法！）



含金量超高的3个学分(学分不好混)

序：本课程关心程序哪个部分？

- 暑假（小学期）刚完成了一个“较大”代码量的作品，现在又要回到“较短”代码的一学期学习？
- 本课的理论课上，不再关注**程序设计**的语法细节，甚至不特别关心用什么**数据结构**（但是，语言和数据结构不过关，算法无法验证，上机会比较麻烦，实验课是“程序设计语言+数据结构+算法”的综合训练）。
- 关心什么呢？

关心程序的核心部分，算法！
如何设计？效率分析？在满足功能的前提下，
计算越快越好，代码越简洁越好！

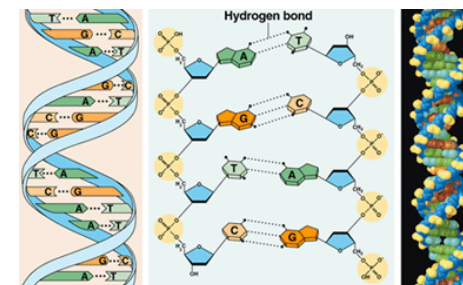
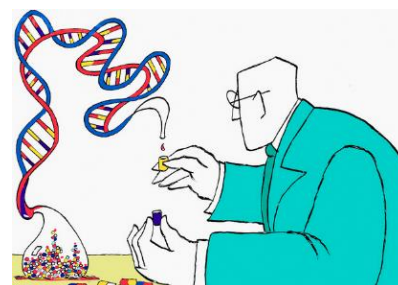
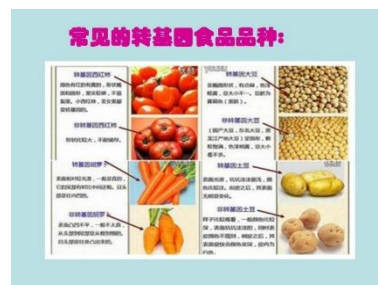
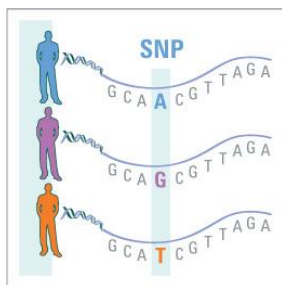
序：算法与数学、编程

- 工程中，通常是代码越长，功能越复杂（代码长、编译时间长、计算时间长，占用空间大），仿佛显得你越牛！
- 算法设计与研究中，代码越短，运行越快，占用资源越少，你越牛！
- 不是数学课，因此不会讲解太多纯演算；
不是编程课，因此不会讲编程语言的技巧等（别期望助教帮你debug）（但编程验证是必要的，可能需要多花时间）。
- 本课对数学、编程要求高；
学好本课，对数学和编程能力有很大提升。

算法研究什么？

例1: Identifying Genes in Human DNA (基因识别)

“生命（物）是四进制”（计算机是二进制）



Identifying all the genes in human DNA（人类基因约 3万个，如何识别并分类？）

determining the **sequences** of the 3 billion(10^9) chemical base pairs that make up human DNA. (a strand of DNA \in finite set {A, C, G, T}) （30亿个碱基对组成人类DNA，如何界定这些碱基对序列，从而进行基因识别？不同碱基对序列组成了不同的DNA。抽象为计算机的排序问题！）

- Computer: 4G Hz CPU, 4×10^9 B/s, suppose that it executes one billion (10^9) instructions per second (设计算速度为：10亿条基本指令/s , [一条指令含多个字节])
- Input size: $n = 3 \times 10^9$
- Insertion sort: running time $T(n) = n^2$ （完成该计算需要的基本指令条数）

$t = s/v$:

$$\frac{3 \times 10^9 \times 3 \times 10^9 \text{ instruc}}{10^9 \text{ instruc/s}} = 9 \times 10^9 \text{ seconds} = \frac{9 \times 10^9}{60 \times 60 \times 24 \times 365} \text{ y} \approx 285.39 \text{ years}$$

例1: Identifying Genes in Human DNA (基因识别)

Identifying all the 100,000 genes in human DNA

determining the sequences of the 3 billion(10^9) chemical base pairs that make up human DNA.

➤ **Insertion sort: running time $T(n) = n^2$**

$$\frac{3 \times 10^9 \times 3 \times 10^9 \text{instruc}}{10^9 \text{instruc/s}} = 9 \times 10^9 \text{seconds} = \frac{9 \times 10^9}{60 \times 60 \times 24 \times 365} \text{y} \approx \underline{285.39 \text{years}}$$

VS

➤ **Merge sort: running time $T(n) = n \lg n$**

$$\frac{3 \times 10^9 \times \lg(3 \times 10^9) \text{instruc}}{10^9 \text{instruc/s}} = 3 \times \lg(3 \times 10^9) \text{seconds} \approx \underline{94.45 \text{seconds}}$$

- ✓ The both sort methods are feasible in reason (两种算法都可行)
- ✓ But insertion sort is impractical (插入排序实际不可行)

例2: Fibonacci Number (斐波那契数)

$$F(n) = F(n-1) + F(n-2)$$

A1: recurrence

```
f(n)
{
    if(n <= 2)
        return 1;
    else
        return f(n-1)+f(n-2);
}
```

$$T_1(n) = ?$$

A2: non-recurrence

```
f1 = 1, f2 = 1;
for(i = 3; i <= n; i++)
{
    f = f1 + f2;
    f2 = f1;
    f1 = f;
}
```

$$T_2(n) = ?$$

例3: Hanoi Tower (汉诺塔)

```
hanoi(n, x, y, z)
{
    if(n == 1)
        x to z;
    else
        hanoi(n-1, x, z, y);
        x to z;
        hanoi(n-1, y, x, z);
}
```

$T(n) = ?$

更多实例...

- 淘宝网 (www.taobao.com)

时间	注册用户数	每日活跃人数	在线商品数	年成交额（亿元）	备注
2003	?			0.34	成立
2005				80	超越沃尔玛
2006		900万			
2007				400	
2009				2083	
2012					双11，1天卖191亿（加天猫），创世界记录
2014	5亿	1.2亿	10亿		双11，1天卖571亿（加天猫），再创记录
2015	? 亿	? 亿	? 亿		双11，1天卖 912 亿（加天猫），创记录？
2017~2024	?	?	?		?

产品分类？不同类型产品成交金额分析？消费行为分析？物流配送？

大数据量稀疏矩阵的SVD（奇异值分解）算法 $R=U \cdot S \cdot V'$

- 天猫、余额宝、支付宝、头条、京东、各个平台的短视频、小红书、.....

更多实例...



2017.6

更多实例...

- 2017年9月, 30万
- 2018年9月, 95万
- 2019年9月, 182万
- 2020年9月, 274万
- 2021年9月, 357万
- 2022年9月, 456万
- 2023年9月, 551万
- 2024年9月, 629万



所有评测记录

序号	用户	题目ID	结果	得分	语言	代码长度 (Bytes)	运行时间 (ms)
4565108		5515464					
4565107		5515463					
4565106		5515462					
4565105		5515461					
4565104		5515460					
4565103		5515459					
4565102		5858					
4565101		5902	Accepted				

2022年
8月29日
456万

2023年
9月4日
551万

ONLINE JUDGE ADMIN 小组 题目 赛事 评测记录 个人信息

所有评测记录

序号	用户	题目ID	结果	得分	语言	代码长度 (Bytes)	运行时间 (ms)
6291715		7785	Accepted	1	c	344	2
6291714		7600	Accepted	1	c	268	3
6291713		7881	Accepted	1	c	944	28
6291712		7881	Compile Error	0	c	1888	0
6291711		7893	Accepted	1	c	444	3
6291710		7893	Wrong Answer	0.666667	c	451	3
6291709		7893	Wrong Answer	0.666667	c	449	5

2024年
9月2日
629万

更多实例...

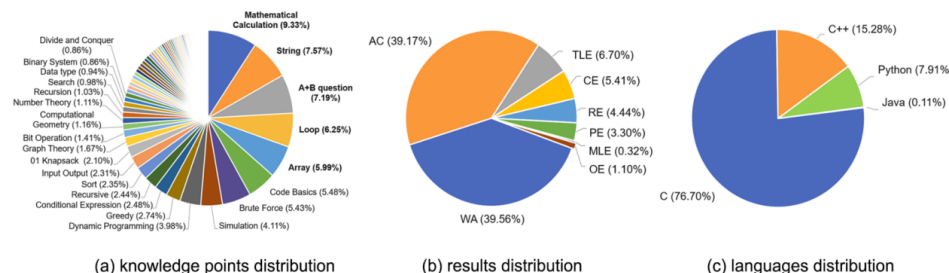


Figure 5. Statistics of the ACcoding dataset.

成果发表在 nature - scientific data上 (2024年5月)


<https://www.nature.com/articles/s41597-024-03392-z/>

更多实例...

- 全国居民身份证管理系统: $n = 1.3 \times 10^9$ 人
- 国家安全防护指纹识别系统: $n \geq 1.3 \times 10^9$ 人
- $2^{67} - 1 = A \times B$
- 天气预报、天文学、.....
- 一个输入输出测试例程与数据 (查找)
- 某市的核酸数据600 PB (1 TB = 1024 GB, 1 PB = 1024 TB)
- 某市的交通数据 > EB量级 (1 EB = 1024 PB, 1 ZB = 1024 EB)

互联网时代...

- 云计算
- 大数据
- 数据挖掘
- 模式识别
- 虚拟现实
- 机器学习
- 人工智能
- 区块链
- 元宇宙
- 大模型



**本质上，
这些研究都是
算法研究！**

课程性质

- 授课对象：本科生，研究生
- 课程性质：base & core in computing（计算领域的基础课、核心课）
- 先导课：程序设计、数据结构、离散、数分、代数、概率

参考书

Introduction to Algorithms (Third Edition), 算法导论(原书第3版), T. Cormen, C. Leiserson, R. Rivest (2002, Turing Award), C. Stein, The MIT Press



¥74.00

算法导论 (原书第3版) 计算机科学 全球
超过50万人阅读的算法圣经!

700+条评价

润知天下图书专营店



每满100减30 (8.26-9.1)

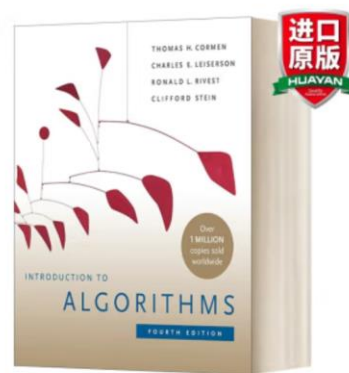
¥105.60

算法导论 (原书第3版) /计算机科学丛书
科技IT引领前沿,计算机科普医学等好书畅

12万+条评价

机械工业出版社

自营



算法导论第四版英文版IntroductiontoAlgorithmsfourthedition精装英

京东价 ¥1400.00 降价通知

增值业务 助力环保,传递知识,旧书换新

配送至 北京海淀区八里庄街道 有货

支持 可配送全球 晚发赔 7天价保 退换货运费险

店铺单笔订单不满48元,收运费8元

由 星梦图书专营店 发货,并提供售后服务. 现在下单,承诺9月5日发货

1

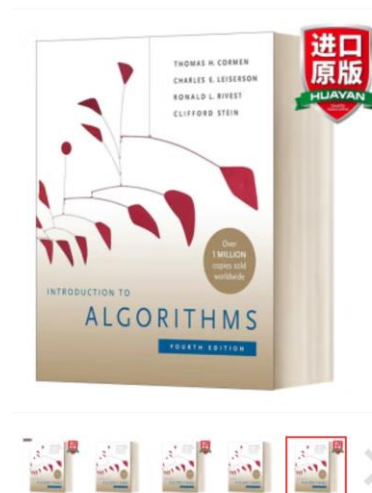
加入购物车

温馨提示 支持7天无理由退货



如果你很有钱, 喜欢原版教材, 也有选择。当然, 内容是一样的。网上看到有第四版的原版。

如果你很喜欢读电子书



“你已经是21系的人了，该学会自己上网找资料了”

更深入的算法学习书籍

《The Art of Computer Programming》 ,
Donald E. Knuth (1974, Turing Award)

更深入的算法学习书籍

网友：“没有读过《Intro...》，不能算是一个真正的程序员”

“计算机算法的圣经”

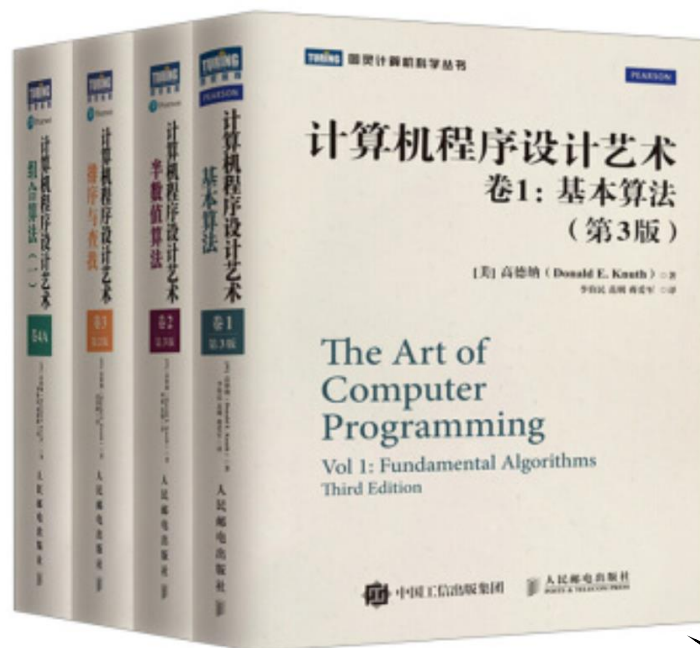
《Introduction to Algorithms》

《The Art of Computer Programming》,
Donald Knuth (1974, Turing Award)

Bill Gates: “如果你认为你是一名真正优秀的程序员，请读Knuth的《计算机程序设计艺术》，如果你能读懂整套书的话，请给我发一份你的简历。”

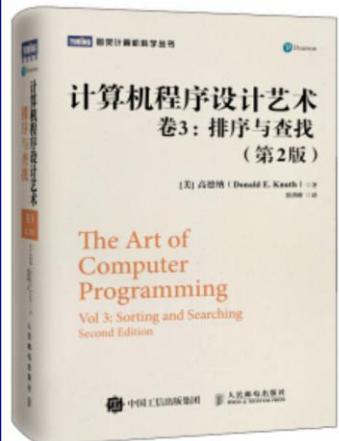
“计算机程序设计理论的荷马史诗”

《The Art of Computer Programming》, Donald E, Knuth (1974, Turing Award)



Bill Gates: “如果你认为你是一名真正优秀的程序员，请读Knuth的《计算机程序设计艺术》，如果你能读懂整套书的话，请给我发一份你的简历。”


“计算机程序设计理论的荷马史诗”



计算机程序设计艺术 卷3: 排序与查找 (第2版)


少年引领科技,科技引领未来.IT/科普/医学/建筑/工业农林每满
高德纳 (Donald, E., Knuth) 著, 贾洪峰 译

京东价 **¥161.80** [8.2折] [定价 ~~¥198.00~~] (降价)

增值业务  礼品包装

配 送 至 有货
由 京东 发货, 并提供售后服务. 23:10前下单, 次日可送达

重 量 1.47kg

服务支持  放心购 闪电退款 | 上门换新 | 破损包赔
京尊达 京准达 自提 49元免基础运费



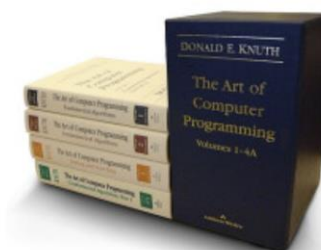
¥3305.00

预订 Art of Computer Programming, The, Volumes 1-4B, Boxed Set (Art of

0条评价

五星店铺 阅世界原版图书专营店

每满99-10



北京无货

¥2387.00

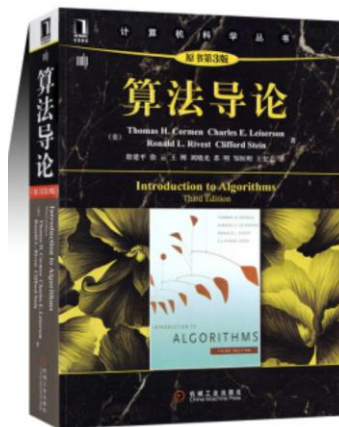
预订 英文原版 The Art of Computer Programming, Volumes 1-4a Boxed Set

10条评价

经济图书专营店

如何读这本书

- 需要仔细品味
- 需要安静思考
- 需要认真推演
- 需要互相讨论



本教材是一本优秀教材！

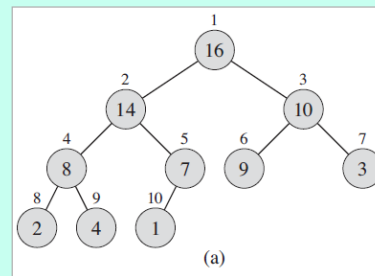
**不但书很厚，而且文字还很多！
不太好读。英文版可能好读些！**

- 跟班上课之中读完授课内容所有章节（如果不上课，自己可能很难静心去读这本书，而且，读起来可能也很难）；适当读参考文献（经典的比流行的生命力更长久）；把上课章节多读几遍更好；读完整本书更好；整本书多读几遍更好；最好把《程序设计艺术》(The Art of Computer Programming)也读了；

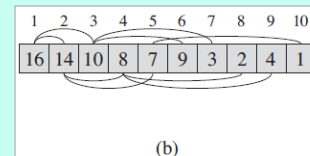
学习方式



工程问题



物理模型



数学模型

- 课堂讲解：建模，分析，原理，来源
- **上机实践**：基本习题和经典习题的上机实践（上机很重要，但不要当成编程课）
- 要能实现算法，会编程实现验证（理论课与上机课鸿沟，理论+实践来解决）
- 只会编程实现还不够，会分析，会推导，会证明，甚至提出问题
- 工程项目中的算法应用，算法改进，算法设计
- 实际的项目训练：在以后的整个软件工程实践中，注重程序设计的可行性，程序的效率，软件测试的性能分析，

学习方式

- 听课
- 看书
- 做作业
- **上机编程**【上机是双刃剑，不要因为上机忽略了推导和分析】
- 课外创作
- 做科研（项目、实验、论文）

不同的阶段做不同事情：如果未来要在专业上走得更远，大二是算法实践（看似刷题但绝不是刷题）的最佳时间，刷题应是主业（这种刷题绝对不是应试教育）！不要等到大四or研三时，才不得不过来疯狂刷题。（而其他同学正在做那个阶段应该做的事）

学习与考核要求

- 课堂要求

学术很自由，课堂很严肃：不迟到、早退；不允许接听电话、大声聊天...

- 学习要求与考核

1. 把所有讲过的算法实现，并进行算法可视化实现，鼓励提交可视化作品。
2. 硬写代码，**刚开始**别用封装完整的库函数，例如：快排不直接用qsort，等，不过，理解了算法后，尽量用封装完整的库函数完成实际的任务。
3. C、C++、Java、Python等任意一种语言实现，甚至matlab等都可以（不过OJ不支持）。
4. 有改进的算法，提供源代码（写上必要注释）、简要的说明文档（使用方法）。
5. *加上灵活可配置参数选项、数据随机产生、基本数据导入导出、可视化效果等更好。
【[ADF WorkShop](#)】【[可视化的排序](#)、可视化的找凸包、.....】
6. 试着写一篇算法相关的小论文。

上机安排、考试安排、计分规则等

- 上机与上机讲评课：见教学日历。
- 考试：只有上机，最后一次大考，平时上机课小考，**练习赛必须全部提交（至少真正 AC 2 题）**。
- 课堂测验与到课，平时上机，期末考试，【题目讲解，大作业，算法研究与应用分享】：

课堂测验与参与【讲解，大作业等】	平时上机	练习赛	期末考试
20%	20%	10%	50%

- 上机计分规则：过题数为主，名次（罚时）为辅，及格线有最低过题门槛。
- 鼓励参与讲评课。
- **不要作弊**：平时上机抄袭作弊容易，考试就被打回原型！有各种技术手段防考试作弊，包括但不限于查重等，然后决定是否采取加试措施。
- **圈定题目，每个人必须提交至少一道题的题解（不低于2页纸）**，内容包括：问题分析、解题思路、数据结构、伪代码、算法分析、应用扩展、启示、等。
- 期末时，每个人交学习总结（可以参加最后两次的总结汇报或演讲）。如果写上机部分，可**参考的思路：考场决策、知识点掌握、编程习惯、思考方向、代码表达方式等等**。
- **大作业：算法可视化、算法应用的完整作品、算法相关的论文、等。**

to: songyou@buaa.edu.cn, chuanluo@buaa.edu.cn cc: TA

不要作弊（不论平时还是考试，都不抄袭别人的代码）

@新京报:【不要作弊的真正原因 😊】加州伯克利大学的教授Brian Harvey在课堂上向他的学生解释为什么不要作弊，不是因为“作弊会对别的学生不公平”，也不是因为“作弊会败坏学校的风气”，而是作弊最终会把你困在一个自己不擅长也不喜欢的职业，困住你真正的人生追求...很有说服力的角度👁️🔗YouTube精选的微博视频



靠作弊得高分，不会伤害老师，也不会伤害其他同学，最终只会伤害自己。

特别提示：

不要拷贝粘贴其他同学的代码提交。

“不要作弊，作弊不会伤害老师，不会伤害同学，因作弊而获利，最终会把你困在一个自己不擅长，也不喜欢的职业，困住你真正的人生追求！”



上机平台

Online Judge (OJ): **`accoding.buaa.edu.cn`**

建设需要不断完善，欢迎同学加入到建设队伍中，包括新功能开发、功能改进、维护、内容建设（题库、题解等）、等等。系统的数据很有研究价值，欢迎有兴趣的同学加入研究小组：排行榜、推荐、.....