

“数据结构” 课程大纲

2020秋季学期

清华大学·计算机科学与技术系

教学团队



考评环节

课号	30240184	00240074	
编程作业 (PA)	24 x3次	18 x4次	尽早阅读OJ首页上的 帮助材料 ： https://dsa.cs.tsinghua.edu.cn/oj/static/pa-book.html 了解 作业形式、相关纪律 ，以及重要的 基本技术 通过网络学堂发布的 注册码 ，在OJ上 加入课堂
期中考试	0	0	基本知识点的充分掌握；基本方法和技巧的灵活运用
期末考试	28	28	
参与	0	0	以课堂参与为主，请及时记录到网络学堂的讨论区
加分	5	5	作业创意 + （讲义、教材、PA）勘误 + 测例设计 + 进步幅度

**** 视疫情的后续演化态势，上述考核方式可能会做**进一步**调整**

基础知识

❖ DSA涉及多个学科，但**并不意味着**必须首先逐一**精通**，常用部分只是其中不大的**子集**

❖ C/C++语言程序设计：类，继承，重载，重写，虚方法，模板

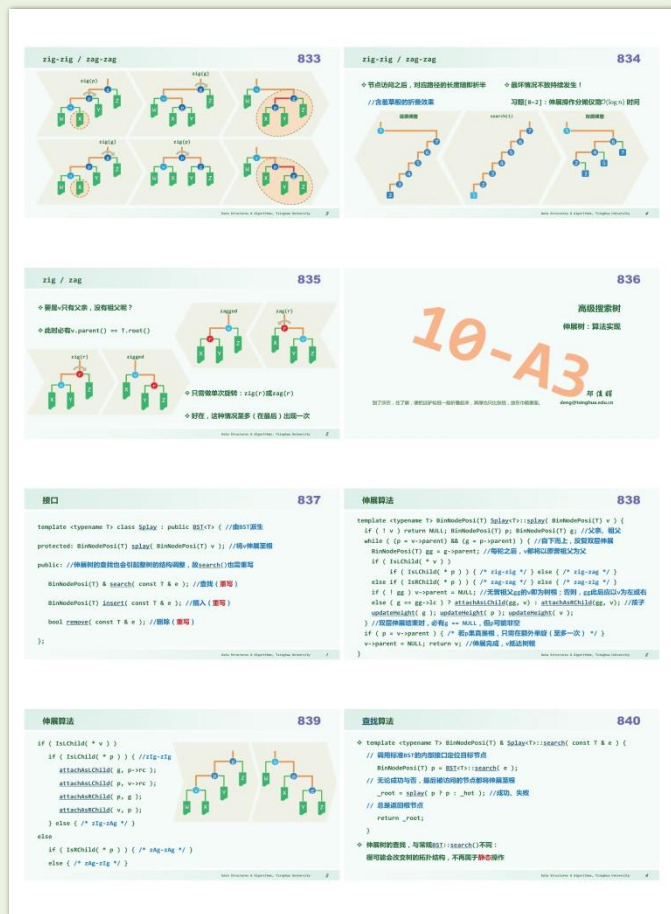
❖ 离散数学：集合，偏序集，良序，数学归纳法；

级数，递归，递推；排列，组合；Stirling逼近

❖ 概率：随机分布，概率，数学期望，期望值的线性率，几何分布

❖ ...

❖ 关于数学，我们会尽可能使用**初等**的方法



按章节独立文档；提供超链接索引，适用于电子阅读器

电子
版

纸
质
版



多种板式，目录统一；彩色、黑白打印两相宜；考试可能用到

学堂在线：<https://www.xuetangx.com/search?query=邓俊辉>

- ✓ 数据结构（上、下）：12章100余节，共500余段视频
- ✓ 计算几何：8章88节，共400余段视频
- ✓ 覆盖校内课堂的大部分讲授内容，是重要的学习辅助渠道，在特殊情况下更有作用
- ✓ 与校内课堂所讲授内容的对应关系，详见打印版讲义目录中的标注
- ✓ 视频之间有大量Quiz，虽难度不大，仍有助于确认自学效果

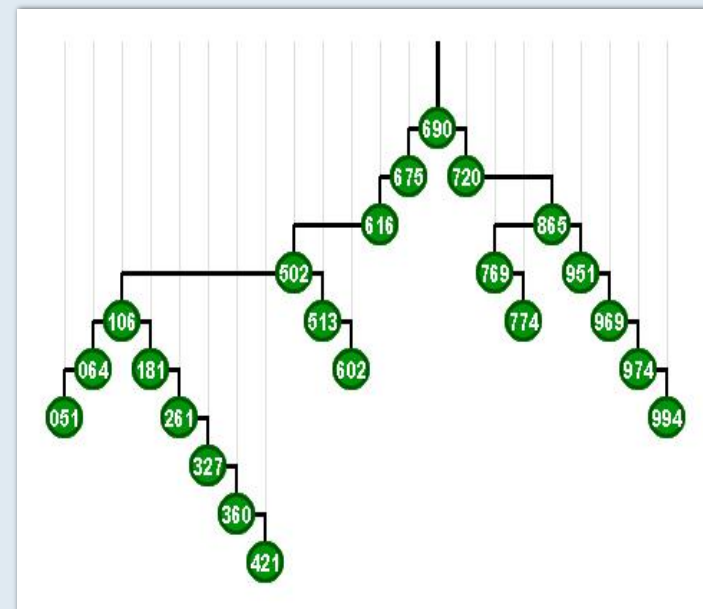
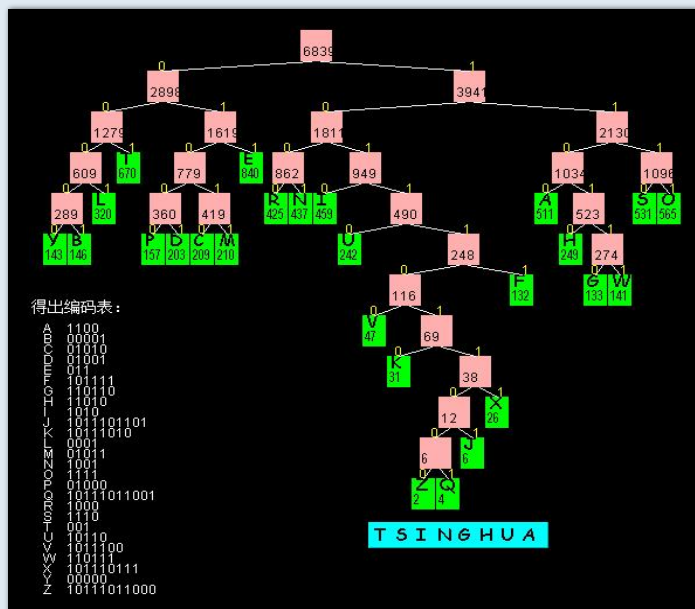
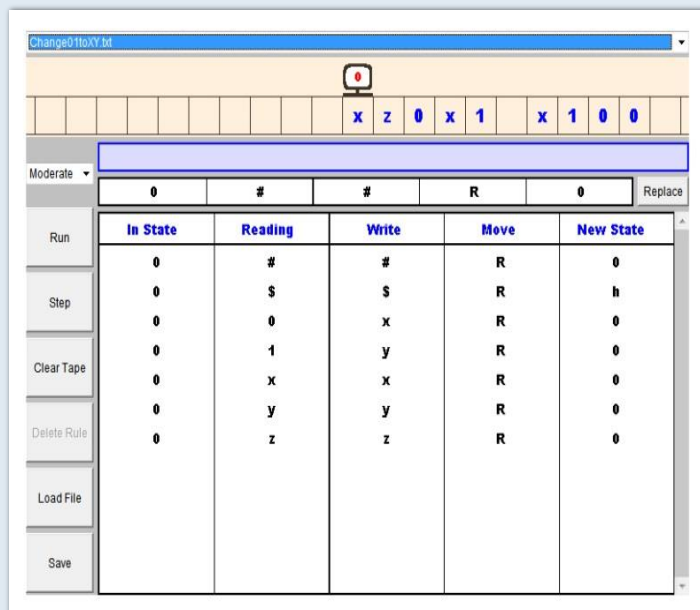
数据结构 (C++) : <https://dsa.cs.tsinghua.edu.cn/~deng/ds/dsacpp/>

- ✓ 内容是讲义的子集，讲解详细但深度不够，校内学生**不建议**购买，可主要依靠讲义
- ✓ 习题解析：针对讲授内容落实巩固，基本部分针对考试，进阶部分着眼拓展，**值得**购买
- ✓ 出版十年来已发现大量错误，阅读前请下载**勘误表**，并对照更正
- ✓ 如有发现新的错误，欢迎报告到网络学堂讨论区的“**勘误报告**”专区

本课程相关的多数**学习**资料，均可经由**教材主页**上对应的链接，从清华网盘**自由**下载

教学演示（网页版）

- ✓ 针对经典的数据结构与算法，通过网络学堂打包发布
- ✓ 多为Java Applet形式，可借助AppletViewer播放（可将index.applet与之绑定）



- ✓ 在老式浏览器中，也可以直接访问：<http://DSA.cs.tsinghua.edu.cn/~deng/ds/DEMO>

教学演示 (Excel版)

✓ 通过网络学堂打包发布：用Excel打开后，随机或手工设置输入，即可反复按F9步进

	A	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	AA	AB	AC	AD	AE	AF	AG	AH	AI	AJ	AK	AL	AM	AN	AO	AP	AQ	AR	AS	
1	<input checked="" type="checkbox"/> Run/Reset	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40				
2	Increments	1	2	3	4	6	8	9	12	16	18	24	27	32	36	48	54	64	72	81	96	108	128	144	162	192	216	243	256	288	324	384	432	486	512	576	648	728	768	864	972				
3																																													
4	Reset	1> Input your increment sequence; 2> Uncheck/check the checkbox to run/reset; 3> Repeatedly press F9 to step forward																																											
5		0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	
6	Sequence	45	289	942	919	898	915	784	290	766	904	460	628	282	719	675	119	679	805	572	518	213	895	191	612	218	384	55	813	168	512	872	160	879	999	391	772	598	495	346	82	60	550	437	

[illegible]

	J	Y	T	V	T	Z	C	N	C	B	W	E	A	C	U	N
S	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
K	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Z	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1
Z	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1
N	0	0	0	0	0	0	1	1	2	2	2	2	2	2	2	2
M	0	0	0	0	0	0	1	1	2	2	2	2	2	2	2	2
E	0	0	0	0	0	0	1	1	2	2	2	2	3	3	3	3
X	0	0	0	0	0	0	1	1	2	2	2	2	3	3	3	3
C	0	0	0	0	0	0	1	2	2	3	3	3	3	3	4	4
M	0	0	0	0	0	0	1	2	2	3	3	3	3	3	4	4
O	0	0	0	0	0	0	1	2	2	3	3	3	3	3	4	4
O	0	0	0	0	0	0	1	2	2	3	3	3	3	3	4	4
M	0	0	0	0	0	0	1	2	2	3	3	3	3	3	4	4
G	0	0	0	0	0	0	1	2	2	3	3	3	3	3	4	4
N	0	0	0	0	0	0	1	2	3	3	3	3	3	3	4	4
T	0	0	0	1	1	1	1	2	3	3	3	3	3	3	4	4

示例代码

- ✓ Visual Studio 2015格式的60个工程
- ✓ 多通过命令行设置输入，直接编译执行
- ✓ 声明后可用于PA，但要小心可能含有的BUG

```
Searching for 0277 ... Not found
Inserting < 0090, W> ... Done
class Skiplist<int,char>[2291816]*12*123:
class Quadlist<struct Entry<int,char> >[3137904]*001: -----
class Quadlist<struct Entry<int,char> >[3137856]*001: -----
class Quadlist<struct Entry<int,char> >[3137808]*001: -----
class Quadlist<struct Entry<int,char> >[3137760]*001: -----
class Quadlist<struct Entry<int,char> >[3137712]*002: -----
class Quadlist<struct Entry<int,char> >[3137664]*005: -----
class Quadlist<struct Entry<int,char> >[3137616]*005: -----
class Quadlist<struct Entry<int,char> >[2955464]*006: -----
class Quadlist<struct Entry<int,char> >[2955416]*016: -----
class Quadlist<struct Entry<int,char> >[2955320]*030: -----
class Quadlist<struct Entry<int,char> >[2954248]*061: -< 0002: K>-----< 0007: M>
class Quadlist<struct Entry<int,char> >[2954032]*123: -< 0002: K>--< 0003: I>--< 0004: F>--< 0007: Q>--< 0007: M>
```

```
Searching for 1455 ... Not found
Removing 3314 ... Not exists
Searching for 0612 ... Not found
Inserting 1783 ... Done
class BTree<int>[4128208]*1:
1783 *>| 1783
```

```
Searching for 2822 ... Not found
Searching for 1668 ... Not found
Searching for 3469 ... Not found
Inserting 3644 ... Done
class BTree<int>[4128208]*2:
3644 *>| 3644
1783 *>| 1783
```

```
Inserting 0178 ... Done
class BTree<int>[4128208]*3:
3644 *>| 3644
1783 *>| 1783
0178 *>| 0178
```

```
Inserting 2307 ... Done
class BTree<int>[4128208]*4:
3644 *>| 3644
2307 *>| 2307
1783 *>| 1783
0178 *>| 0178
```

```
Removing 1957 ... Not exists < 0016: I>---
Searching for 0190 ... Not found < 0016: I>---
Inserting 1798 ... Done < 0016: I>---
class BTree<int>[4128208]*5: < 0016: I>---
3644 *>| 3644 < 0016: I>---
2307 *>| 2307 < 0016: I>---
1798 *>| 1798 < 0012: H>---< 0012: A>---< 0016: I>---
1783 *>| 1783
0178 *>| 0178
```

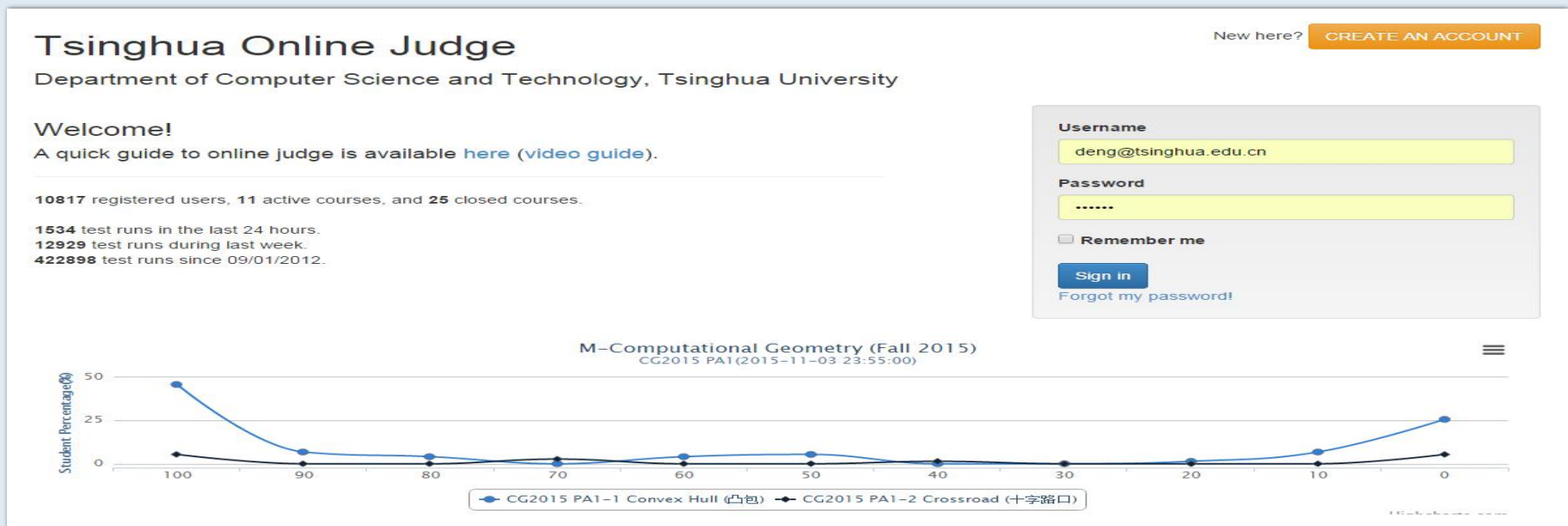
```
Inserting 1839 ... Done
class BTree<int>[4128208]*6:
```

```
3644 *>| 3644
2307 *>| 2307
```

Online Judge (OJ)

先用email注册加入平台，再从网络学堂发放的邀请码加入课堂

务必认真阅读首页上的帮助文档、编程作业说明



<http://DSA.cs.tsinghua.edu.cn/OJ>

Programming Assignment (PA)

- ❖ 须**独立完成** 否则后果**严重**
双方**同论**，不作区分；无论是否同班、同系、同届
- ❖ 如何判定？ 代码**雷同度**
- ❖ 标准、经典算法**除外**： 二分查找、KMP、Dijkstra、快速排序、DFS，...
- ❖ 什么**不可以**？ 源代码，或足以导致雷同的伪代码
- ❖ 什么**可以**交流？ 题意理解、解题思路
算法、数据结构的设计与选用方案
测试用例（欢迎**设计**，并随同**文档**提交到网络学堂**答疑区**）
- ❖ 尺度拿捏不准？ 涉及的人员、文献、资源等，须在**HonorCode**中标注
只要注明得客观、准确、完整，就至少不会倒扣