

Welcome to ICS

元培数科 王恩博

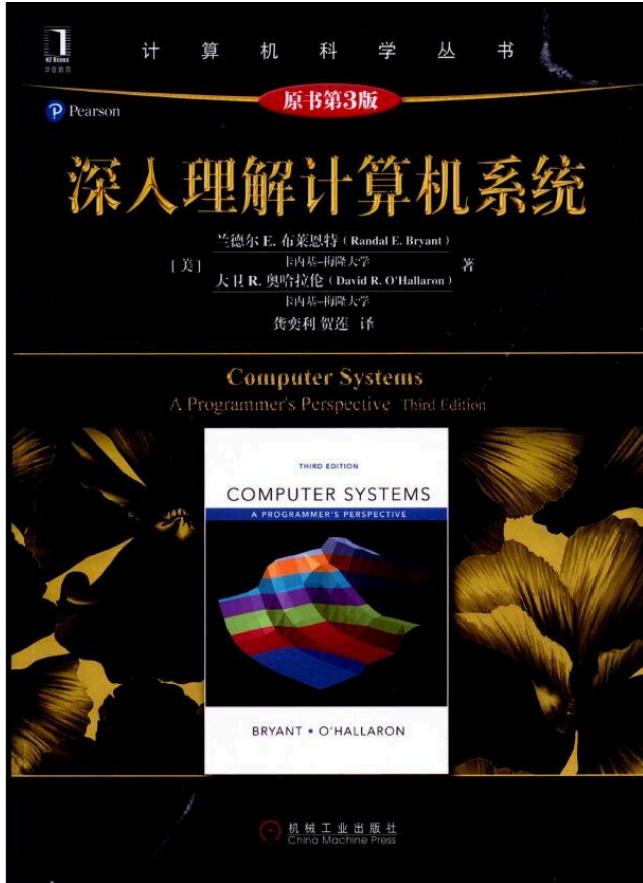
Let's get started →

2025/9/10



目录

- 1. Welcome to ICS
- 2. 助教介绍
- 3. 自我介绍
- 4. 老师介绍
- 5. 小班安排
- 6. 小班安排
- 7. 小班安排
- 8. 分数评定
- 9. Lab 安排
- 10. Lab 要求
- 11. Lab 要求
- 12. Lab 要求
- 13. Lab 目标
- 14. 环境说明
- 15. 环境说明
- 16. 可用资源
- 17. 个人建议
- 18. 基础知识
- 19. 基础知识
- 20. 基础知识
- 21. THANKS





群聊: ICS 小班 17 (25秋)



该二维码 7 天内 (9月17日前) 有效, 重新进入将更新

**关注教学网！
关注学号邮箱！**

助教介绍

王恩博

- 院系：元培学院
- 专业：数据科学与大数据技术
- 年级：大三
- 邮箱：webrun@stu.pku.edu.cn
- 微信：[13269431217](#)
- 喜欢旅行、听音乐、+CS
- 有任何问题欢迎发邮件/微信
- 很期待和大家一起学习！

自我介绍

- 姓名、专业、年级
- 兴趣爱好
- 对 ICS 这门课的了解
- 如果不是专业必修为什么选了 ICS 这门课
- anything else
- 建议 1min 左右

龚鹤宁

陈泽羽

孔令珊

刘筱

张哲涵

黎祖含

陈俊宇

胡勋炀

黎宣萱

李准珩

陈俊言

胡思成

王欣语

曾若天

老师介绍

王亚沙

- 北京大学博雅特聘教授，软件工程国家工程研究中心主任，教育部长江学者，中国计算机学会杰出会员，全国数据标准化技术委员会数据技术标准工作组组长。长期从事人工智能、数据挖掘及医信交叉研究。担任国家重点研发计划项目负责人，并承担国家多项自然科学基金、国家863计划、“核高基”重大科技专项课题。研究成果获国家科技进步二等奖、国家教育部科技进步奖一等奖。
- [王亚沙-北大学者主页 \(scholar.pku.edu.cn\)](http://scholar.pku.edu.cn)

小班安排

周次	日期	大班课	主题	日期	小班课	日期	大班课	主题	LAB
	周一	5~6节		周三	10~11节	周四	5~6节		
第1周	9月8日	1	Overview	9月10日	1	9月11日	2	Bits and Bytes/Integers	L1 data (周四)
第2周	9月15日	3	Floating Point	9月17日	2	9月18日	4	Machine Prog: Basics	
第3周	9月22日	5	Machine Prog: Control	9月24日	3	9月25日	6	Machine Prog: Procedures	L2 bomb (周一)
第4周	9月29日	7	Machine Prog: Data	10月1日	国庆放假	10月2日	国庆放假		
第5周	10月6日	国庆放假		10月8日	国庆放假	10月9日	8	Machine Prog: Advanced	L3 attack (周四)
第6周	10月13日	9	阶段测验1 (第1~8课)	10月15日	4	10月16日	10	Processor Arch: ISA&Logic	
第7周	10月20日	11	Processor Arch: Sequential	10月22日	5	10月23日	12	Processor Arch: Pipelined	L4 arch (周一)
第8周	10月27日	13	The Memory Hierarchy	10月29日	6	10月30日	14	Cache Memories	L5 cache (周四)
第9周	11月3日	15	Program optimization	11月5日	7	11月6日	16	Linking	
第10周	11月10日	17	ECF: Exceptions & Processes	11月12日	8	11月13日	18	ECF: Signals & Nonlocal Jumps	L6 tsh (周四)
第11周	11月17日	19	阶段测验2 (第10~18课)	11月19日	9	11月20日	20	System Level I/O	
第12周	11月24日	21	Virtual Memory: Concepts	11月26日	10	11月27日	22	专题讲座	
第13周	12月1日	23	Virtual Memory: Systems	12月3日	11	12月4日	24	Dynamic Memory Allocation	L7 malloc (周四)
第14周	12月8日	25	Network Programming I	12月10日	12	12月11日	26	Network Programming II	L8 proxy (周四)
第15周	12月15日	27	LAB测验 (L1~L7)	12月17日	13	12月18日	28	Concurrent Programming	
第16周	12月22日	29	Synchronization: Basic	12月24日	14	12月25日	30	Synchronization: Advanced	
考试周	12月29日		期末考试 (周一下午)						

小班安排

周次	日期	大班课	展示同学	日期	对应大班节数	日期	大班课	展示同学
	周一	5~6节		周三	10~11节	周四	5~6节	
第1周	9月8日	1		9月10日	1	9月11日	2	龚鹤宁 陈泽羽
第2周	9月15日	3	孔令珊 刘筱	9月17日	2/3	9月18日	4	张哲涵 黎祖含
第3周	9月22日	5	陈俊宇 胡勋炀	9月24日	4/5	9月25日	6	黎宣萱 李準珩
第4周	9月29日	7	陈俊言 胡思成	10月1日	国庆放假	10月2日	国庆放假	
第5周	10月6日	国庆放假		10月8日	国庆放假	10月9日	8	王欣语 曾若天
第6周	10月13日	9	阶段测验1 (第1~8课)	10月15日	6/7/8	10月16日	10	龚鹤宁 陈泽羽
第7周	10月20日	11	孔令珊 刘筱	10月22日	10/11	10月23日	12	张哲涵 黎祖含
第8周	10月27日	13	陈俊宇 胡勋炀	10月29日	12/13	10月30日	14	黎宣萱 李準珩
第9周	11月3日	15	陈俊言 胡思成	11月5日	14/15	11月6日	16	王欣语 曾若天
第10周	11月10日	17	龚鹤宁 陈泽羽	11月12日	16/17	11月13日	18	孔令珊 刘筱 张哲涵 黎祖含
第11周	11月17日	19	阶段测验2 (第10~18课)	11月19日	18	11月20日	20	陈俊宇 胡勋炀
第12周	11月24日	21	黎宣萱 李準珩	11月26日	20/21	11月27日	22	陈俊言 胡思成
第13周	12月1日	23	王欣语 曾若天	12月3日	22/23	12月4日	24	龚鹤宁 陈泽羽
第14周	12月8日	25	孔令珊 刘筱	12月10日	24/25	12月11日	26	张哲涵 黎祖含 陈俊宇 胡勋炀
第15周	12月15日	27	LAB测验 (L1~L7)	12月17日	26	12月18日	28	黎宣萱 李準珩
第16周	12月22日	29	陈俊言 胡思成	12月24日	28/29	12月25日	30	
考试周	12月29日		期末考试 (周一下午)					

小班安排

- 研讨题按照上表进行，每人每次一道大班研讨题
- 暂定一人负责一题，后续可能根据实际情况调整
- 每人展示时长 6-10 min, 原则上不超过 15 min
- 非负责同学也要积极提前思考准备
- 展示后其他同学可以进行提问并进行讨论
- 展示的质量和研讨的积极度会作为小班评分依据
(之一)
- 另外根据进度和时间情况可能进行习题练习、LAB 介绍、LAB 复习、知识复习、考试习题讲解等内容

- **研讨题**

- 题目例子：暂时不能给你明确的答复

- **研讨形式**

- 如果使用PPT（不必须）周三中午12:00前发送至班级微信群、北大网盘中
 - PPT命名方式：`W2-Float-Q1-王恩博.pptx`
 - 若有额外补充材料，可同步发送在微信群中
 - 展示完成后同学互评（问卷星）+老师助教评分
 - 0-10 分打分，低于 6 分需解释原因

分数评定

考勤要求 (如需请假需提前说明)

研讨质量

小班参与

车斤
自

Lab 安排

程序与数据

Data Lab

汇编语言

Bomb Lab/ Attack Lab

处理器体系结构/程序性能

Arch Lab

分级存储体系

Cache Lab

虚拟内存

Malloc Lab

异常控制流

Shell Lab

网络、并发

Proxy Lab

Lab 要求

- 本学期共设置 8 个 lab，原则上每个 lab 持续 14 天
- lab 的发布、评分均在 autolab 平台：<http://162.105.31.232/>
- **尽早开始**
 - 避免最后突击，出现各种 bug 但 de 不出来逾期
 - **今年没有 grace day!**
- **仔细阅读要求**
 - 注意代码风格
 - 从 Arch Lab 开始，可能有助教手动评分的部分
 - 避免 magic number，文件/函数需要注释
- **认真、独立完成**
 - 有查重机制，严禁抄袭（网络、往年代码），可以参考课本代码
 - 建议使用 git、github 进行版本管理
 - 但不要把自己的代码放到 public 的仓库里

Lab 要求

		Name	Section	Rank	min#	备注
	2	25		30	325	25 float64_f2完全一致
	15	298		2	25	25 代码逻辑有调整不建议认定作弊
	22	24		2	25	24 float64_t2i完全一致
	10	147		2	25	25 float64_f2完全一致
	22	24		30	325	24 float64_f2完全一致
	8	204		27	329	204 虽然多处一致，但均为简单题，难以避免，不认定为抄袭
	30	362		24	260	260 连续多个函数仅大小写、缩进不一致
	18	365		6	361	361 float_i2f完全一致
	17	191		2	25	25 float64_f2完全一致
	22	24		17	191	24 float64_f2完全一致
	8	2		22	24	2 float_i2f逻辑一致
	6	361		5	19	19 float_i2f仅大小写不一致
	23	47		9	357	47 float_i2f, float64_f2完全一致
	5	19		1	75	19 float_i2f大小写替换
	23	47		20	44	44 satMul3大小写替换
	30	362		23	47	47 satMul3大小写替换
	28	264		20	44	44 satMul3大小写替换
	23	47		28	264	47 satMul3大小写替换
	14	226		21	22	22 countTrailingZero缩进替换
	25	236		5	19	19 float_i2f一致

Lab 要求

```
请根据要求补全函数/*
* bitAnd - x&y using only ~ and |
* Example: bitAnd(6, 5) = 4
* Legal ops: ~ |
* Max ops: 8
* Rating: 1
*/
int bitAnd(int x, int y) {
    return
}
```

要解决这个问题，我们需要仅使用 `~`（按位非）和 `|`（按位或）操作符来实现 `x & y`（按位与）的功能。

根据德摩根定律，我们知道：`x & y = ~(~x | ~y)`

这个公式的原理是：

1. 对 `x` 和 `y` 分别进行按位非操作 (`~x` 和 `~y`)

bitAnd.c

```
1  /*
2  * bitAnd - x&y using only ~ and |
3  * Example: bitAnd(6, 5) = 4
4  * Legal ops: ~ |
5  * Max ops: 8
6  * Rating: 1
7  */
8  int bitAnd(int x, int y) {
9      return ~(~x | ~y);
10 }
11
```

Lab 目标

- Data Lab: 掌握整型与浮点型的存储方式与特性，熟悉部分位运算技巧
- Bomb Lab: 掌握阅读二进制可执行程序与汇编代码的方法
- Attack Lab: 了解栈溢出的原理，利用该原理攻击风险程序
- Arch Lab: 了解 HCL 语言，掌握流水线冒险机制，并以此优化 Y86 指令实现以及流水线结构
- Cache Lab: 掌握缓存机制，以此优化内存访问顺序
- Shell Lab: 掌握文件系统和 I/O，熟悉异常处理，实现简易 shell 终端
- Malloc Lab: 掌握内存分配机制，实现一个动态内存分配器
- Proxy Lab: 了解网络通信，熟悉并发方法，实现简易的网络代理

根据往年经验，Arch Lab、Shell Lab、Malloc Lab 三个任务较为困难，请留出充分的时间完成

环境说明

Clab

- Lab环境配置与说明: [PKU Clab](#)、[Clab 快速开始](#)、[Clab&ICS](#)、[VScode-SSH](#)

The screenshot shows the PKU Clab website. At the top, there is a navigation bar with links for 文档 (Documentation), 博客 (Blog), 通知 (Notifications), 关于 (About), and a search icon. Below the navigation bar, the title "云计算实验平台 (试运行)" is displayed, followed by a subtitle "(试运行)". A note below the title says "开箱即用的免费云计算服务, 为教学与科研服务". There are several icons for different services or tools. Below these are two buttons: "登录使用 >" and "入门文档 >". At the bottom, there are four cards describing the platform's features:

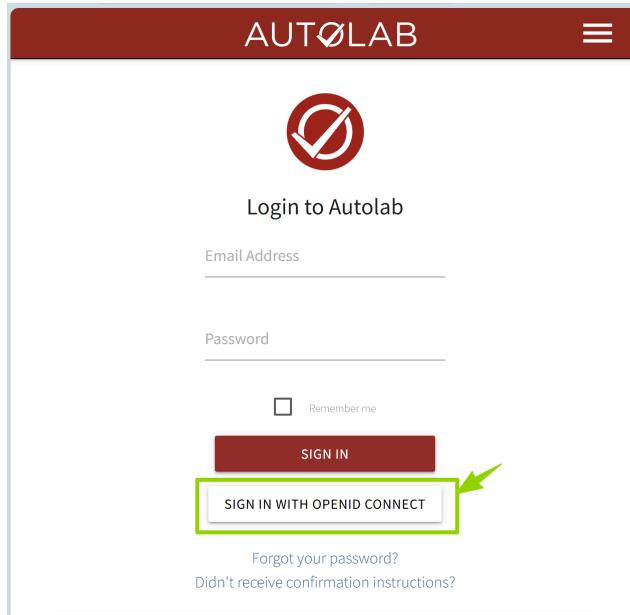
- 环境丰富**: 提供各种 Linux 发行版镜像, 预配置了许多开箱即用的环境。
- 配置简单**: 简单易用的界面, 轻松配置您的开发环境。
- 性能优异**: 采用最新的硬件, 确保您的服务有足够的性能。
- 稳定可靠**: 采用多副本技术, 确保服务的稳定性和可靠性。

- pku-new : 直接访问网络
- sudo apt update &&sudo apt upgrade :记得时常更新
- sudo systemctl status lcpud : 超过30min, 90%cpu占用, 直接kill掉
- clabcli connect : 连接网关
- clabcli disconnect : 断开网关

环境说明

Autolab

直接访问地址: (<http://162.105.31.232/>) , 课程码 LAM3V1



- 根据网站管理的要求, Autolab后续需要同学们自行设置域名解析以访问。
- 同学们需要编辑使用设备(笔记本、CLab虚拟机等)的hosts文件并添加如下内容:

162.105.31.232 autolab.pku.edu.cn

- 修改hosts文件可参考下面的说明: [Windows](#)、[Linux \(CLab\)](#)、[Mac](#)

可用资源

或许你会更想了解一些可能有用的链接.....

北大网盘

2023Fall-PKU

Tag: ics · Arthals' ink

ICS Memos

LCPU Getting Started

csdiy

CSAPP Errata

一些课程资料

北京大学 2023 年秋计算机系统导论课程 (ICS) 作业、笔记、经验

更适合北大宝宝体质的Lab讲解

答疑平台

Linux 俱乐部维护的新手教程

CS自学指南：大学纯靠自学

CSAPP 勘误

个人建议

■ 作为一门5学分硬课，想必大家对任务量和难度有所耳闻

■ **教材：**理解+细节

1. 上完这门课，理论上至少需要通读两遍（无论是否预习）：课后+作业+Lab 前看一遍，考试之前看一遍
2. 有的时候遇到卡壳，请继续往下读，你~~的问题很有可能在后面被解决；可以借助 AI？但有可能胡说八道~~
3. 中文版有的时候有问题，看英文原版（特别是第7章链接）以及勘误。

■ **Lab：**被玩命魔改的恐怖作业

1. 网络资料？AI？**有Lab小测**(但可以帮助debug)
2. Lab 小测的形式？统一上机（暂时不能给你明确的答复）
3. 首先独立思考并尝试完成，如果真的想不出来可以借鉴往年思路，但切忌~~眷抄~~
4. Lab的任何具体方法不会在DDL之前答疑，但有其他问题可以随时来问

■ **考试与往年题：**往年题库充分

1. 规划好期中各阶段测试、期末的时间（并不是一个能速通的课）
2. 最开始不会做、做的时间很长很正常。熟能生巧（包括知识和做题方法两方面）24期末
3. 遇到不理解答案的题可以尝试把题目和答案一起喂给 AI 尝试，很可能提供有用的思路。助教时间有限

基础知识

Linux指令

如果你没有基础，现在可以开始学习了：常用 Linux 命令

ls

列出文件和目录

cd

切换目录

mkdir

创建目录

rm

删除文件或目录

mv

移动文件

cp (-r)

拷贝文件 (夹)

基础知识

Linux指令

chmod

修改权限

sudo

以 root 权限执行命令

su

切换用户

ps

进程管理

vim/nano

编辑文件/Ubuntu自带

scp

远程拷贝

基础知识

Something else

- 你可能还需要了解一下 Git 和 GitHub, SSH 等, 这些东西在你未来的学习中会经常用到
 - 一些中文版的简单参考Cithub官方、Linux、Windows
 - 连接虚拟机使用的 SSH 也可以了解, 如果出现报错可以完整复制参考 AI 解决方法
 - (注意连接 Clab 机器需要北大校园网, 在外需要 VPN)
 - LCPU Getting Started

强烈推荐找些博客或者 B 站/YouTube 视频学习一下它们! (当然也可以结合 AI 随用随学)

THANKS

Made by WEB-05

webrun@stu.pku.edu.cn

Reference: [WalkerCH]'s and [Arthals]'s presentations.



扫一扫上面的二维码图案，加我为朋友。