

数值计算方法实训

上海电力大学

数理学院数学系

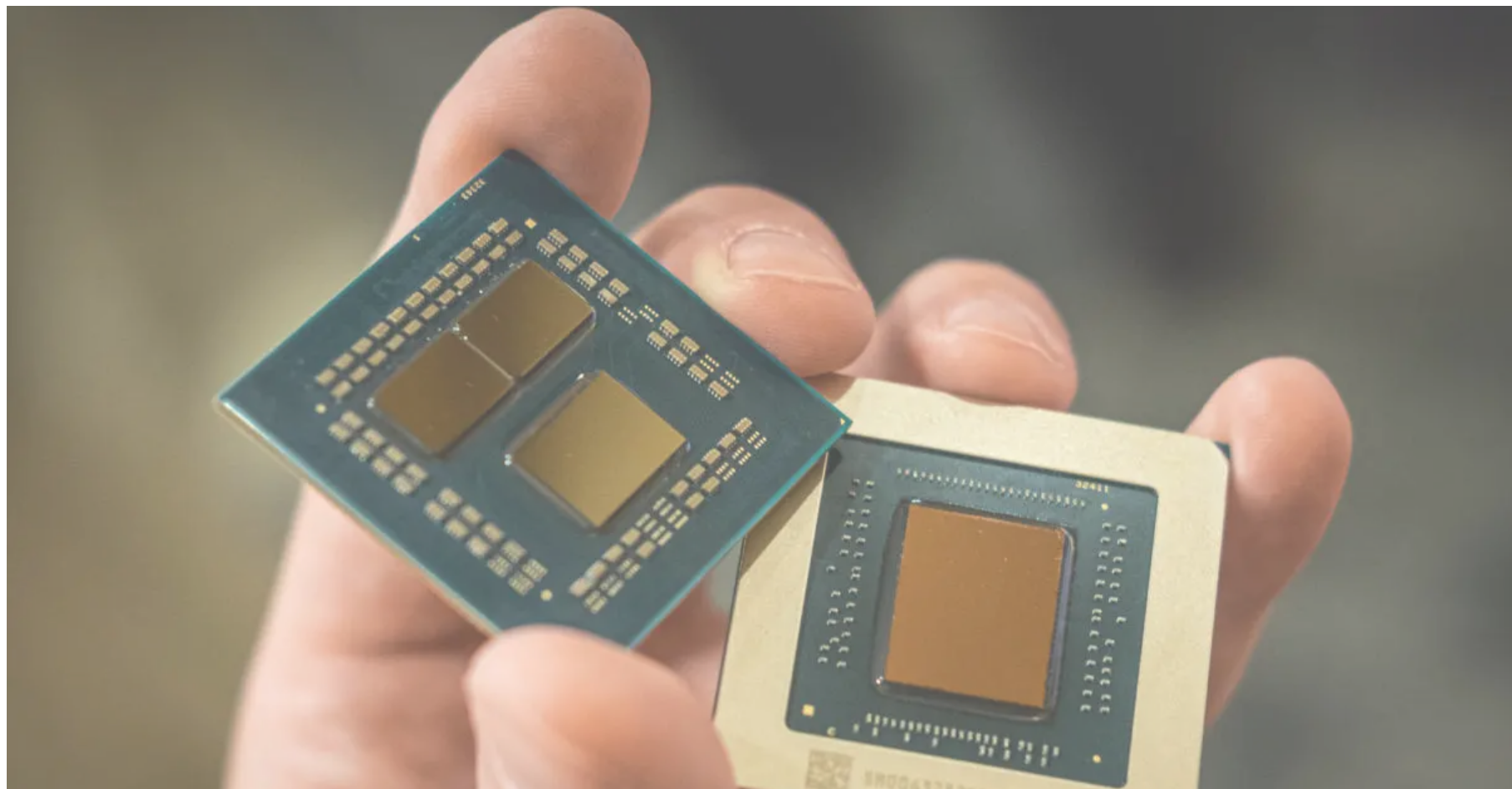
邓化宇



课程信息

(What)什么是“数值计算”?

从一颗芯片开始



(Why)为什么要学“任何东西”?

为什么要学“任何东西”？

为什么要学Python/数值计算实训？

- 为什么要学微积分/高等代数/XXXX/...？
- 长辈/学长：擦干泪不要问为什么

学过微积分以后，再看为什么要学微积分

微积分的几个重要主题

- 启蒙、应用与挑战（Newton时代）
 - 机械论世界观（模型驱动的系统分析）
 - 数学是理解世界的“基本工具”：导数、微积分基本定理、...
- 严格化与公理化（Cauchy时代）
 - 各种卡出的bug（Weierstrass函数、Peano曲线...）
- 大规模问题的数值计算（Von Neumann时代）
 - 优化、有限元、PID...
 - AI是未来人类社会的“基本工具”
- 三个主题应该根据学科特点各有侧重
 - 感受：学了很多，但好像都没学懂

为什么要学“任何东西”？

重走从~~无~~到~~有~~的发现历程

- 基本思想、基本方法、里程碑、走过的弯路
- 最终的目的：应用/创新（做题得分不是目的而是手段）
 - 如果只是记得几个结论，那ChatGPT已经做得很好了

学习“任何东西”的现代方法

- 使用辅助工具加速探索
 - 数值/符号计算：numpy,sympy,sage,Mathematica,...
 - 可视化：matplotlib
 - All-in-one: Jupyter (2017 ACM Software System Award)
 - 人生苦短，Python是岸。

(How)怎么学“任何东西”?

Prerequisites

信息与计算科学专业学生必须具备的核心素质。

1. 是一个合格的计算机用户

- 会STFW/RTFM自己动手解决问题
- 不怕学习和使用任何可以提高效率的工具
 - Linux,VS Code,Python,...

2. 不怕写代码

- 能管理一定规模(数千行)的代码
- 能在出bug时默念“机器永远是对的、我肯定能调试出来的”
 - 然后开始用正确的工具/方法调试

学术诚信（Academic Integrity）

Academic Integrity 不是底线，而是“自发的要求”

- 对“不应该做的事情”有清楚的认识
 - 不将代码上传到互联网
 - 主动不参考别人完成的实验代码
 - 不使用他人测试用例(depends)

有些行为可能使你得到分数，但失去应有的训练

成为一名计算机的Power User

感到 Linux/PowerShell/Python/...很难?

1. 没有建立信心、没有理解基本逻辑
 - 计算机科学自学指南
2. 没有找对材料/没有多问“能不能再多做点什么”
 - Baidu v.s. Google/Github/Stackoverflow v.s. ChatGPT
3. 没有用对工具(该用VS Code就别用Vim/Emacs)
 - 过了入门阶段，都会好起来

成为一名计算机的Power User

- 如果一件事很困难，通常是因为没有找到正确的方法。
- 这门课程中绝大部分的内容都可以在互联网上找到适当的学习资料。
- 在互联网/搜索引擎普及的时代，只要掌握“提出问题”的能力，就能极快地提升自己。
- 同时，这可能将在 AI 时代被颠覆——AI 可能会猜测你的目的，并且告诉你“应该去做什么”。

学会写代码

写代码 == 创造有趣的东西

- 命令行 + 编程语言就是全世界
 - 我们还有sympy, sage, z3, rich,...呢
- 不要讲语言特性、设计模式、...
 - 写代码就对了；你自然而然就会需要它们的，也就自然而然学会了。

Thanks