

# 陆子易

求职意向: 软件工程师

22岁 男 上海 应届本科留学生

电话: +1-510-495-7161

微信: neutrino\_iv

邮箱: [lzy\\_1006@icloud.com](mailto:lzy_1006@icloud.com)

领英: [linkedin.com/in/luziyi](https://www.linkedin.com/in/luziyi)



## ↳ 教育经历

美国加州大学, 伯克利 UC Berkeley

计算机科学、天体物理 双专业

本科 Bachelor of Arts

绩点: 3.768 / 4.0

2021年12月毕业

相关课程: 软件工程、人工智能、计算机结构、算法、计算机安全、数据结构、信息设备与系统设计、线性代数……

## ↳ 近期项目

Snap!Con, 5人敏捷团队中的 Ruby on Rails 开源活动网站开发, 2021年2月 - 至今

[snapcon.org](https://snapcon.org)

1. Snap!Con 是每半年 300名免费编程教育者和其学生的大会, 也将官网代码作为活动网站框架进行开源
2. 作为扁平化团队中的网络开发者之一, 在 Ruby on Rails 框架中进行全栈测试驱动开发 TDD 和持续整合 CI
3. 近期将外部邮件管理系统的 RESTful API 整合进了元数据更改的队列, 并用RSpec 框架和 WebMock替身测试

敌意存储环境中的 Go 语言安全文件分享系统: 设计与实现, 2020年11月 - 2020年12月

1. 在 2人团队中设计并实现了一个端到端加密的文件储存、分享系统
2. 使用 Go 语言, 并调用给定的密码学和不安全存储 API
3. 设计并实现了多用户文件共享、编辑和选择性的访问权撤回

一些 Ruby on Rails 网络应用练习, 2020年6月 - 2020年8月

[rotten-potatoes-09213.herokuapp.com](https://rotten-potatoes-09213.herokuapp.com)

1. 实现了一些互动性 MVC 网络应用, 如一个 Hangman 拼字游戏和一个电影数据库
2. 使用 Ruby on Rails 框架进行测试驱动开发 TDD, 并持续整合/部署 CI/CD 至 Heroku 云
3. 使用 RSpec 和 Cucumber 库撰写单元和集成测试, 覆盖率均达~92%

高性能 C 语言线性代数模块的实现与优化, 2020年4月 - 2020年5月

1. 以 C 语言实现了一个线性代数模块
2. 用缓存模块化和循环展开等技巧、OpenMP 多线程库和 x86 AVX 底层 SIMD 多数据流指令进行优化
3. 为  $2000 \times 2000$  矩阵的幂函数性能提升约 7 倍

流水线结构的 RISC-V 处理器的数据通路实现, 2020年2月 - 2020年4月

1. 使用 Logisim 在数据通路层面实现了标准设计的流水线结构 RISC-V CPU
2. 使用 RISC-V 汇编为 30 余个指令编写了 40 余条测试
3. 为算术逻辑单元 ALU 实现了 14 条指令

## ↳ 相关技能

1. 语言: Python 3, Ruby, Go, Java, C, RISC-V 架构与汇编
2. 技术: Ruby on Rails 网络应用框架, git 版本管理, Unix 命令行, Heroku 云, NumPy 库, Matplotlib 可视化库
3. 方法: 敏捷开发, 面向对象 OOP, MVC 模式, 测试驱动开发 TDD (RSpec, Cucumber), 持续整合 CI (Travis)
4. 普通话标准, 英语流利 (8年全英文教育经历, GRE 164 + 170 + 4.0, 托福 113)

## ↳ 研究经历

基于八叉树的星系 H-alpha / beta 谱线辐射转移数值模拟, 2019年5月 - 2019年8月

[bitbucket.org/lzy-1006/h-art](https://bitbucket.org/lzy-1006/h-art)

1. 使用八叉树数据结构描述三维空间, 对星系中不同波长光子的随机游走进行蒙特卡洛数值模拟
2. 使用 C 语言扩展了旧模拟程序, 并用 Python 3、NumPy 并行运算库和 Matplotlib 可视化库处理数据
3. 生成并处理了 15GB 的光子数据

## ↳ 工作经历

学生助教 (光学与红外天文学实验课), 2020年8月 - 2020年12月

1. 因实验课成绩优秀, 帮助向 20 名大三、大四生教授光学与红外天文学实验课。每周工作 8 小时
2. 主要在课堂教授包括 Python 的 SIMD 并行科学计算、数据处理工具的使用和易维护的编码习惯
3. 每周开放 2 小时面谈, 经常帮助解决科学编程问题