

Rust 程序设计语言

“锈化”圣经注解

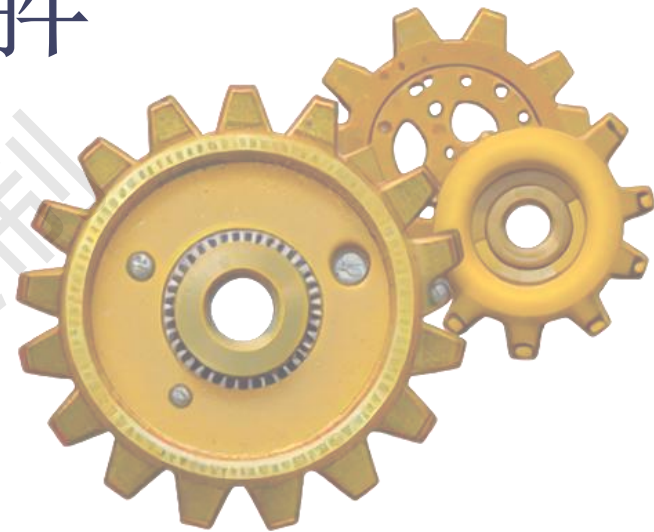
The Rust Programming Language,
comments go through

00 序言



OS 视界

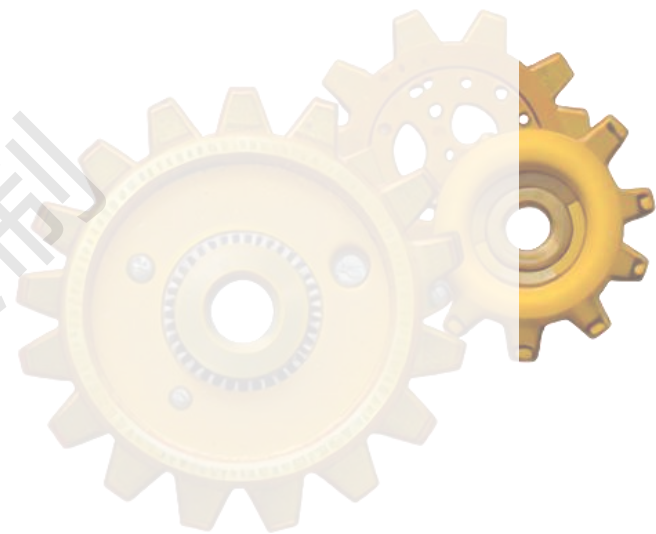
2024.04





目录

- 为什么学习Rust语言
- Rust程序设计语言
- 注者简介
- 内容介绍
- 材料和更新计划
- 建议食用（学习）指南
- 下期预告



为什么学习 Rust 语言

Rust for Linux

Article [Talk](#)

From Wikipedia, the free encyclopedia

Rust for Linux is a series of [patches](#) to the [Linux kernel](#) that adds [Rust](#) as a second [programming language](#) to C for writing kernel components.

History [\[edit\]](#)

The Linux kernel has been primarily written in C and [assembly language](#) since its first release in 1991. Around 1997, the addition of [C++](#) was considered and experimented upon for two weeks before being scrapped.^[1] Rust was created in 2006 and combines the performance of [low-level programming languages](#) (such as C) with a focus on [memory safety](#) and a [user-friendly tool set](#) and [syntax](#).^[2]

The Rust for Linux project was announced in 2020 in the [Linux kernel mailing list](#) with goals of leveraging Rust's memory safety to reduce [bugs](#) when writing [kernel drivers](#).^[3] At the [Open Source Summit 2022](#), [Linus Torvalds](#) stated that the incorporation of the project's work could begin as soon as the Linux 5.20 release, later named as Linux 6.0.^[4] The first release candidate for Linux 6.0 was created on 14 August 2022, without Rust support. In the release notes for Linux 6.0-rc1, Torvalds expressed his intention for adding Rust support, "I actually was hoping that we'd get some of the first rust infrastructure, and the multi-gen LRU VM, but neither of them happened this time around."^{[5][6]} On 19 September 2022, an article from [ZDNet](#) revealed an email from Linus Torvalds stating that "Unless something odd happens, it [Rust] will make it into 6.1".^[7]

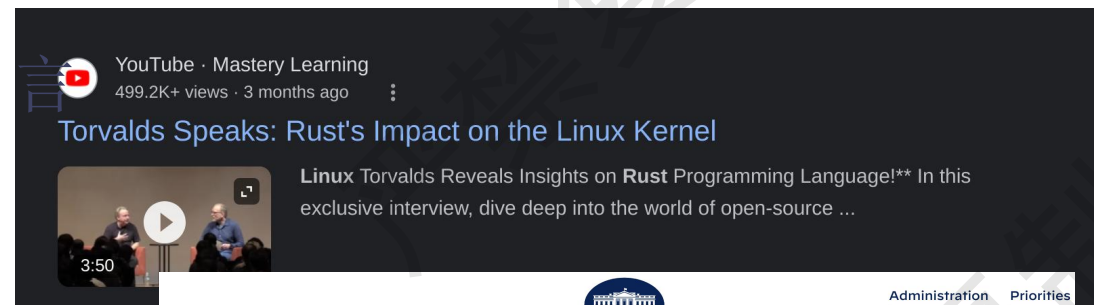
In October 2022, a [pull request](#) for accepting the implementation for Rust for Linux was approved by Torvalds.^[8] As of Linux 6.1, support was intentionally left minimal in order to allow developers to test the feature.^[9]

对此，Linux 之父 Linus Torvalds 也曾发表看法：Linux 最终不会用 Rust 编写，没有人会用 Rust 重写内核的 2500 万行 C，但是他也看到了 Rust 的优势，鼓励采用缓慢但稳定的方法将 Rust 引入 Linux，同时他表示将 Rust 接口用于驱动程序和其他非核心内核程序是有道理的。



Rust for Linux

Developer(s)	Community contributors Miguel Ojeda
Repository	github.com/Rust-for-Linux/linux
Written in	Rust
Available in	English
License	GPL-2.0-only with Linux-syscall-note.
Website	https://rust-for-linux.com/




FEBRUARY 26, 2024

Press Release: Future Software Should Be Memory Safe

[SANTIAGO PASTORINO](#) [@spastorino](#)

"Rust teams at Google are as productive as ones using Go, and more than twice as productive as teams using C++." [@larsberg_](#) [#RustNationUK](#)



12:31 AM · Mar 28, 2024 · 525.9K Views

White House Call to Address Root Cause of Worst Cyber Attacks

[See full report \[here\]\(#\)](#)

House Office of the National Cyber calling on the technical community to be in cyberspace. ONCD makes the case prevent entire classes of vulnerabilities by adopting memory safe programming the research community to address the to enable the development of better irity quality.

[Building Blocks: A Path Toward Secure and](#)

* linux 内核三十多年来的第二种系统开发语言

Rust 程序设计语言

← → ↻ 🔒 https://www.rust-lang.org/zh-CN/learn ☆ ⌵ ⌵ ⌵ ⌵ ⌵



安装 学习 实验 工具 团队 社区 博客 简体中文 (zh-CN)

学习 Rust

开启您的 Rust 之旅

《Rust 程序设计语言》被亲切地称为“圣经”。本书从基本原则出发，给出了 Rust 语言的概览。您将在阅读本书的过程中构建几个项目，读完本书后，您就能扎实地掌握 Rust 语言。

此外，Rustlings 课程会指导您下载并设置 Rust 工具链，在命令行中教您阅读和编写 Rust 代码的语法基础。它可以运行在您自己的环境中，是《通过例子学 Rust》之外的另一种选择。

如果您不喜欢阅读大量的文档来学习语言，那么《通过例子学 Rust》就能涵盖您要学的知识。虽然本书花了很多篇幅来解释代码，但它展示的代码很丰富，并且尽量减少了文字解释。它还包括很多练习！

阅读本书！

阅读非官方翻译

学习 RUSTLINGS 课程！

阅读《通过例子学 RUST》！

阅读非官方翻译

12. Rust 最佳实践

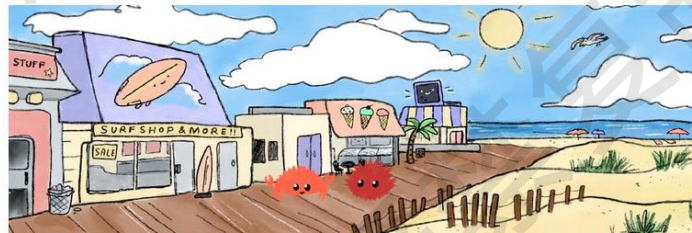
13. 手把手带你实现链表

攻克编译错误

Rust的“圣经”有点多，难？

://course.rs/about-book.html ☆

Rust语言圣经(Rust Course)



Rust 语言真的好：连续八年成为全世界最受欢迎的语言、没有 GC 也无需手动内存管理、性能比肩 C++/C 还能直接调用它们的代码、安全性极高 - 总有公司说使用 Rust 后以前的大部分 bug 都将自动消失、全世界最好的包管理工具 Cargo 等等。但...

有人说：“Rust 太难了，学了也没用”

对于后面一句话我们持保留意见，如果以找工作为标准，那国内环境确实还不好，但如果你想成为更优秀的程序员或者是玩转开源，那 Rust 还真的是不错的选择，具体原因见下一章。

至于 Rust 难学，那正是本书要解决的问题，如果看完后，你觉得没有学会 Rust，可以找我们退款，哦抱歉，这是开源书，那就退 🌟 吧！

如果看到这里，大家觉得这本书的介绍并没有吸引到你，不要立即放弃，强烈建议读一下进入 Rust 编程世界，那里会有不一样的精彩。

配套练习题

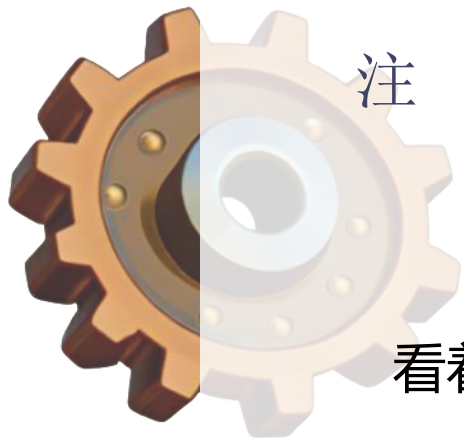
对于学习编程而言，读一篇文章不如做几道练习题，此话虽然夸张，但是也不无道理。既然如此，即读书又做练习题，效果会不会更好？再加上练习题是书本的配套呢？：P

- Rust 语言实战, Rust 语言圣经配套习题，支持中英双语，可以在右上角切换

创作感悟

截至目前，Rust 语言圣经已写了 170 余章，110 余万字，历经 1000 多个小时，每一个章节都是手动写就，没有任何机翻和质量上的妥协(相信深入阅读过的读者都能体会到这一点)。

曾经有读者问：为什么你的书后面章节，写了删删删，删了删删删，只有它(开源圣经)能一直存在



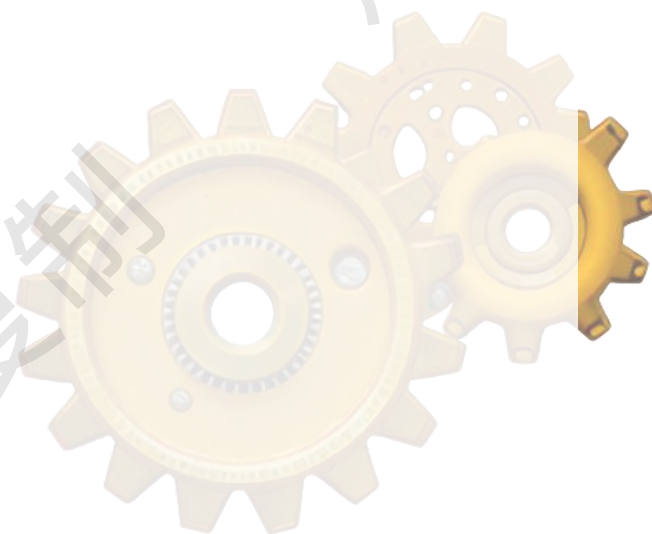
注 者 简 介

看着谭浩强的书，敲着Turbo C长大的C程序员

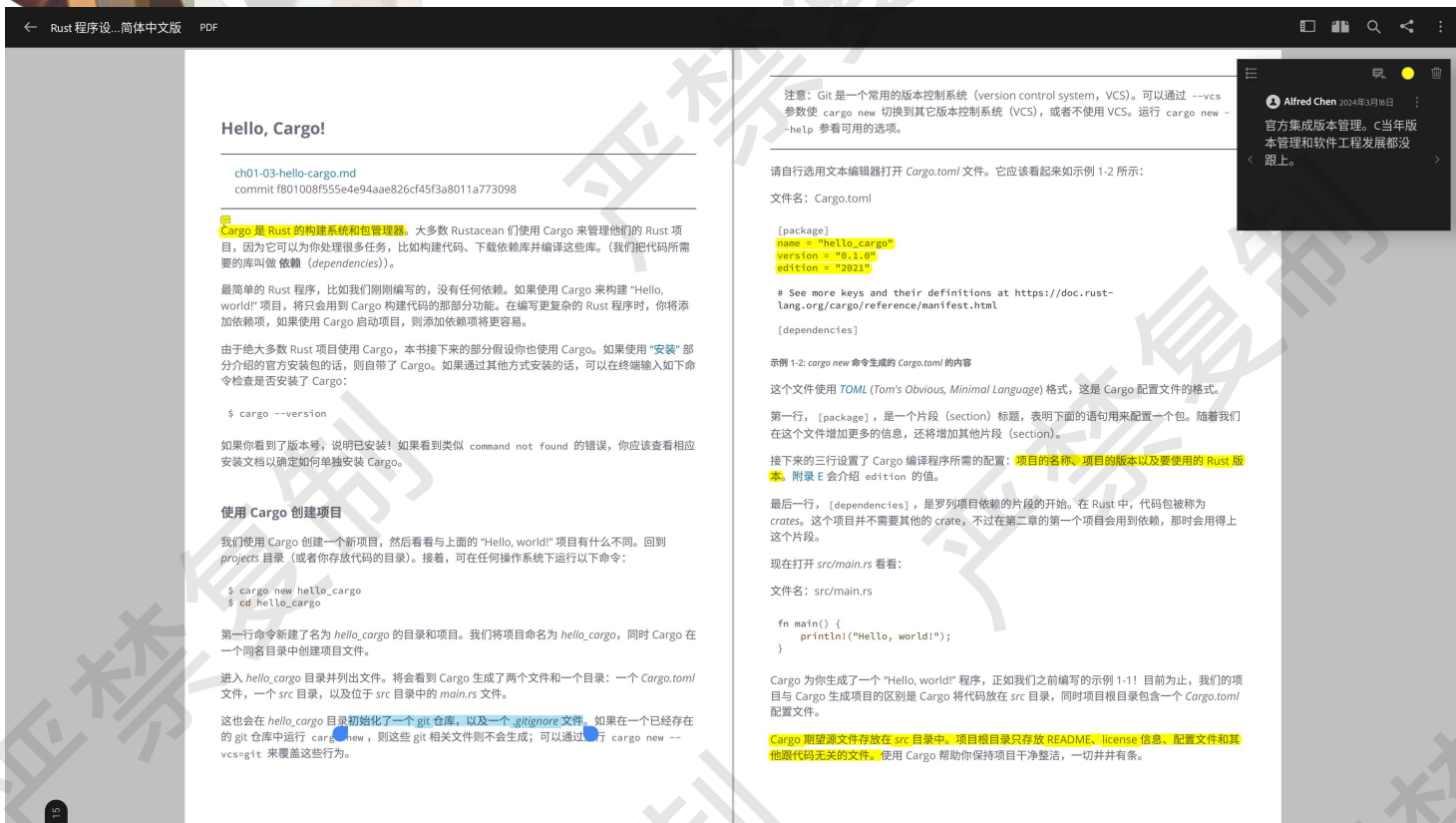
十多年的内核开发者

Project C BMQ/PDS CPU调度器/算法的作者

-- Alfred Chen



内 容 介 绍

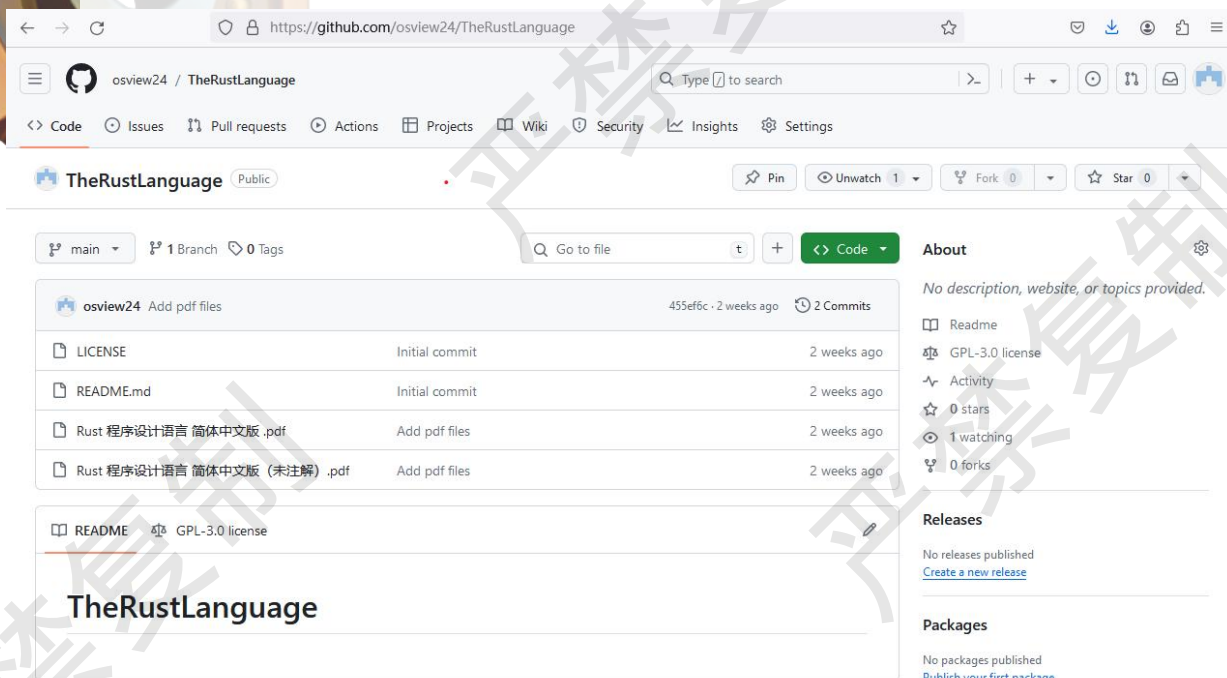


- ***不是*手把手教程**
- **不包含高亮（黄色/蓝色）内容**
- **我们的笔记注释讲解**
- **额外的测试小程序和讲解**

以
最朴素，最直接
的方式告诉你我们对Rust语言
的看法

* 注解/看法谨代表注者当时/视频制作当时的想法

材料和更新计划



材料仓库

<https://github.com/osview24/TheRustLanguage>

注解版Rust程序设计语言 简体中文版(pdf)
未注解版Rust程序设计语言 简体中文版(pdf)
视频ppt的pdf版(如果有的话)

更新计划

暂定**每周一更**，周五晚前更新

每期视频**<10分钟**

更新通告请关注 OS视界 动态



建议食用（学习）方式

1. 细读 未注解版Rust程序设计语言 **简体中文版/英文版 pdf/网页版**

- ◊ 自己的想法
- ◊ 高亮/注释

2. 书中例子的练习

- ◊ 出错信息
- ◊ 自己额外的测试例子

3. **观看我们的注解 (pdf/视频)**

- ◊ 听一下我们的见解


~1小时

最佳辅助学习工具

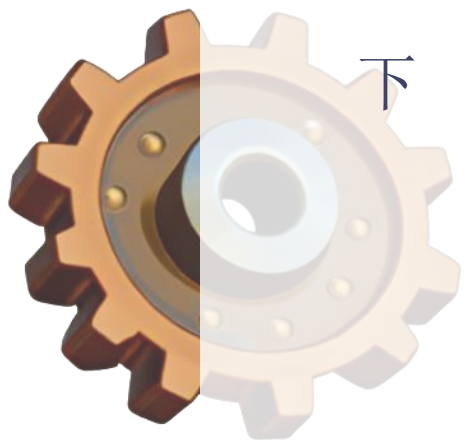
- 人工智能辅助学习

-  任何的AI工具, ChatGPT, Gemini, Kimi...

- 生物科技辅助学习

-  肌肉记忆, 您的双手

* 现阶段我们建议的学习方式



下

期

预

告

覆盖<<Rust程序设计语言>>

- 前言
- 第一章



- 更新不易，求关注，求三连
- 你的支持对我们真的很重要

