

编译原理概述

魏恒峰

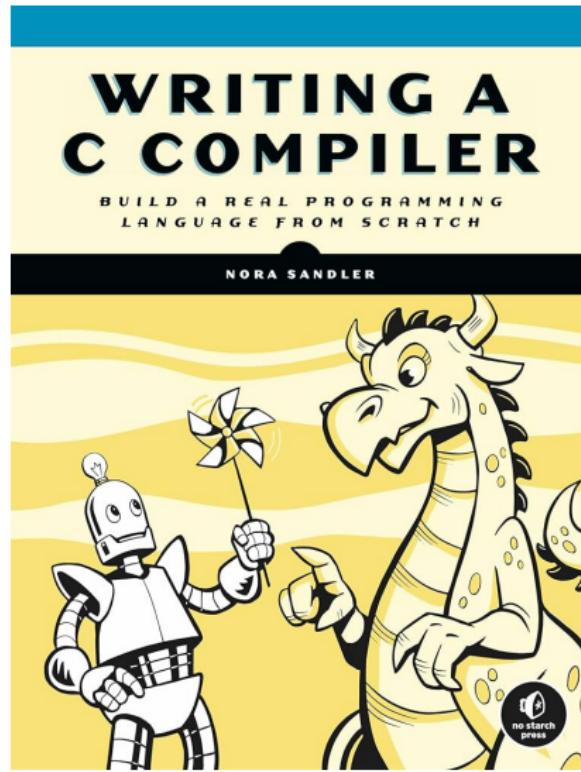
hfwei@nju.edu.cn

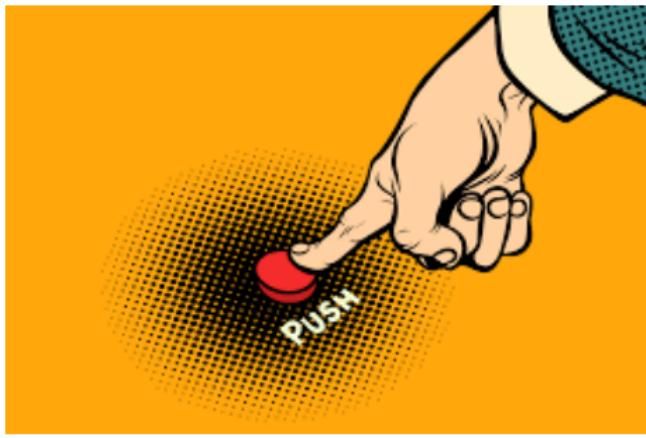
2024 年 03 月 01 日 (周五)





哦？编译原理？

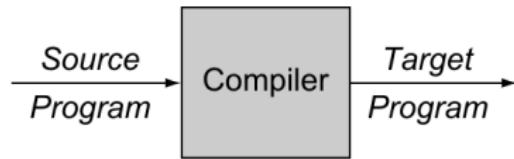




gcc factorial.c -o factorial

“高级”语言 \Rightarrow (通常) “低级”语言 (如, 汇编语言)

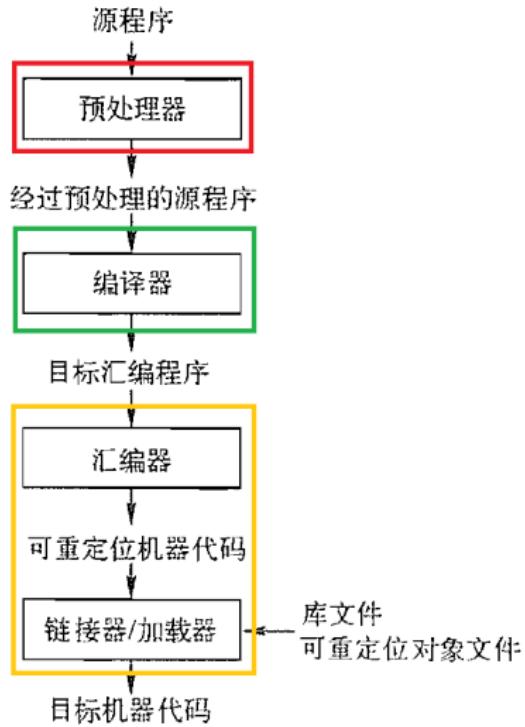
汇编语言经过汇编器生成机器语言



GopherJS - A compiler from Go to JavaScript

godoc reference used by 1.2k projects PASSED

GopherJS compiles Go code (golang.org) to pure JavaScript code. Its main purpose is to give you the opportunity to write front-end code in Go which will still run in all browsers.



Q : 机器语言是如何跑起来的?

作业 (P1 ~ P9): <https://www.bilibili.com/video/BV1EW411u7th>
(计算机科学速成课 40 集全 Crash Course Computer Science)



“我只想安静地做个码农，你为什么要来强迫我？”







Alda is a text-based programming language for music composition. It allows you to write and play back music using only a text editor and the command line.

```
piano:  
o3  
g8 a b > c d e f+ g | a b > c d e f+ g4  
g8 f+ e d c < b a g | f+ e d c < b a g4  
<< g1/>g/>g/b/>d/g
```

The language's design equally favors aesthetics, flexibility and ease of use.

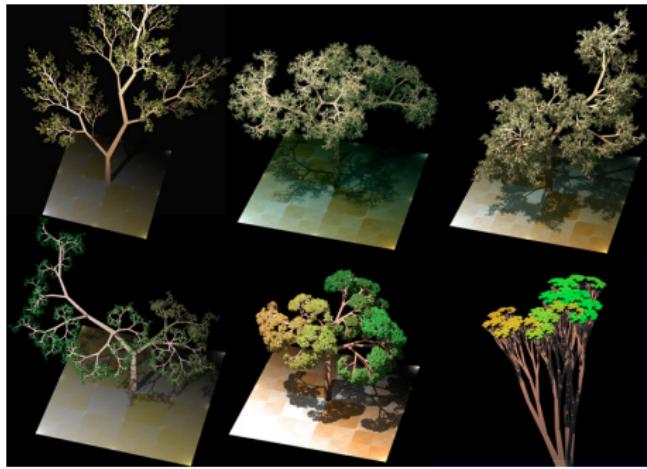
alda repl

alda play -f .alda

alda @ 知乎



10 Demo: Alda @ youtube



Fractal Grower (Try It!)

Sierpinski Triangle @ wiki



语言类应用程序

- ▶ 配置文件解析 (.properties)
- ▶ CSV 文件 (Comma-Separated Values)
- ▶ JSON 文件 (JavaScript Object Notation)
- ▶ SQL 引擎 (Structured Query Language)
- ▶ TLA⁺/TLAPS (TPaxos.tla)
- ▶ (Java) 字节码解释器
- ▶ C/C++ 语言编译器
- ▶ 排版工具 (LATEX)
- ▶ 绘图工具 (TikZ, Dot/Graphviz)
- ▶ L-System (Cantor Set)



学习编译原理最大的难点：“只见树木，不见森林”



絕紙知上
此得事來
要終始覺
行淺



考勤 (0%): 非必要不点名, 不需要请假

平时作业 (0%): ≈ 10 次作业, 每次 ≤ 3 题

课程实验 (60%): 8 ~ 10 次实验

期末测试 (40%): 考试周统一安排; 3 小时; 上机考试

期末测试 (40%): 考试周统一安排; 2 小时; 开卷

每周五发布作业 下周五 23 : 55 前自愿提交作业



邀请码: 8G928EBJ

课程实验：开发 SysY 语言编译器



L0: 环境配置 本周五 18:00 发布

鼓励讨论，但需独立编码完成课程实验



课程实验：抄袭者当次实验计 0 分

QQ 群号: 869910463



助教: 顾龙、李和煦、钱品亦

<http://docs.compilers.cpl.icu/>



Q 搜索

- > 课程简介
- > 课程通知
- > 课程教材
- > 课程作业
- > 课程资源
- > 抄袭与惩罚
- > 课程实验

编译原理课程网站，请**收藏**并及时关注网站更新



hengxin +material/

778270f · 4 minutes ago

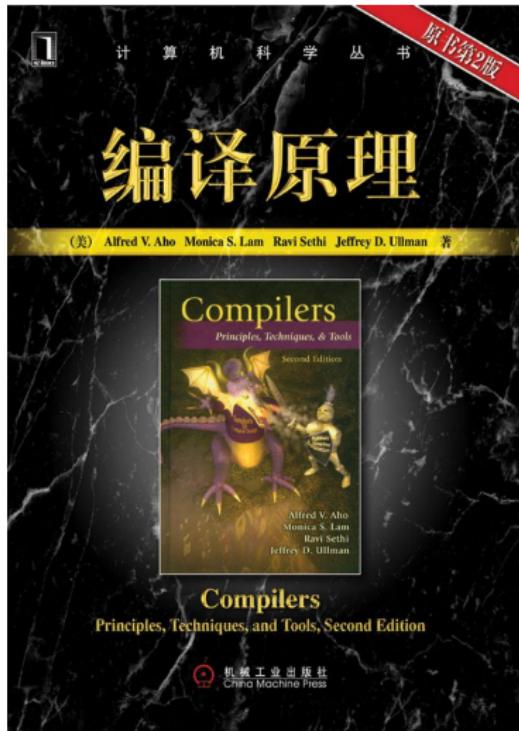
[History](#)

Name	Last commit message	Last commit date
..		
0-overview	+material/	4 minutes ago
material	+material/	4 minutes ago
README.md	+material/	4 minutes ago
preamble.tex	+0-overview	7 minutes ago

<https://github.com/courses-at-nju-by-hfwei/compilers-lectures/tree/master/2024>

overview.pdf

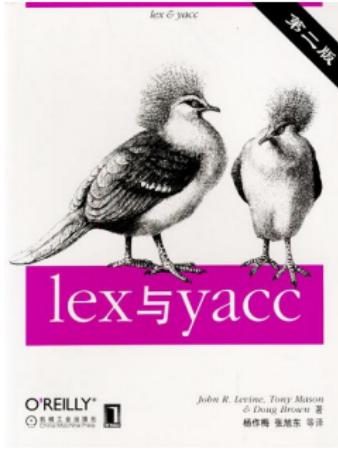
overview-handout.pdf



也可使用“本科教学版”



Flex: 词法分析器生成器



Bison: 语法分析器生成器

不够现代, 本学期课程实验**不再支持**这些工具



(Since 1988)



Terence Parr (University of San Francisco)

<https://www.antlr.org/index.html>

<https://www.antlr.org/tools.html> (IntelliJ Plugin)

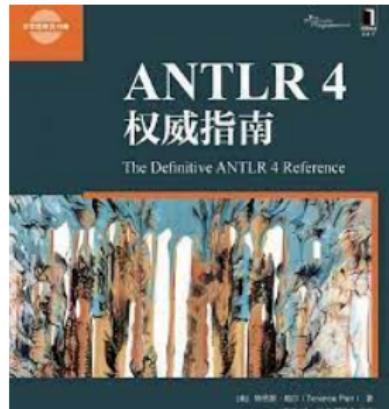
<http://lab.antlr.org/> (Online lab)

The Definitive
ANTLR 4
Reference



Terence Parr

The
Pragmatic
Programmers



基于 ANTLR 4, 是课程实验指导的**重要**参考资料



Language Implementation Patterns

Create Your Own Domain-Specific and General Programming Languages

Selected by Pragmatic Bookshelf Policy

Terence Parr



Language
Implementation Patterns

编程语言 实现模式

Create Your Own Domain-Specific
and General Programming Languages

[译] Terence Parr 著
李直康 译
高鹏翔 审校



基于 ANTLR 3, 与 ANTLR 4 相比有些过时,
但可以看作理解 ANTLR 4 的基础

自制编译器

How to Develop a Compiler

[日] 青木峰郎 / 著 严基连 焦云 / 译

Turing
图灵精英
设计丛书

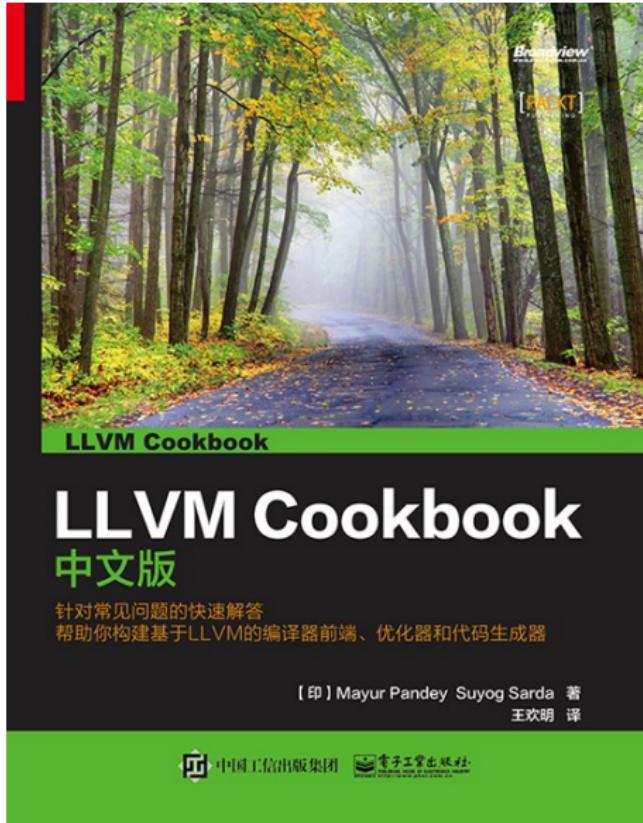


中国工信出版集团



人民邮电出版社
POSTS & TELECOM PRESS

“从零开始制作真正的编译器”，对课程实验很有帮助，**强烈推荐**



从某次实验开始，你就会开始接触 LLVM (<https://llvm.org/>)



LLVM 简介 @ Bilibili

更多参考书, 随课程进展陆续发布



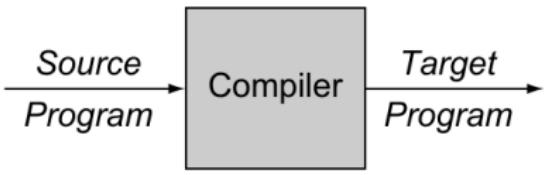
<http://docs.compilers.cpl.icu/#/2024/resources>

资源汇总

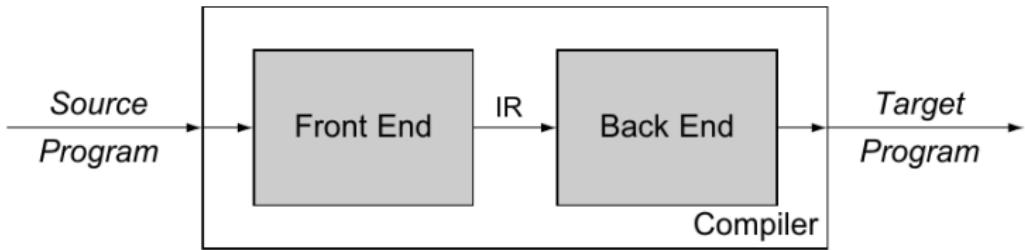
-
- [compilers-resources @ github](#): 编译原理相关资源
 - [books: 电子书籍](#)

LET'S GET
STARTED





IR: Intermediate Representation (中间表示)

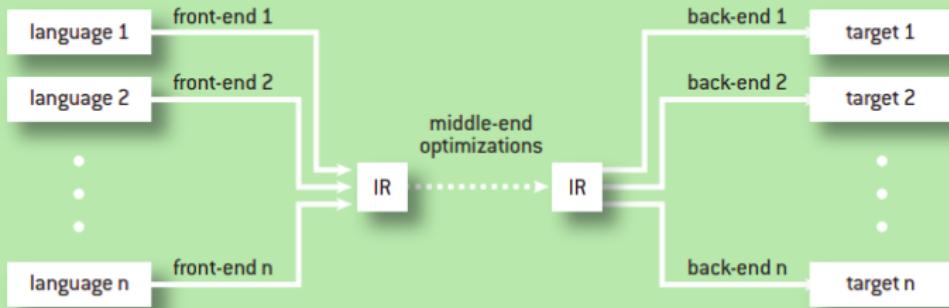


前端（分析阶段）: 分析源语言程序, 收集所有必要的信息

后端（综合阶段）: 利用收集到的信息, 生成目标语言程序

FIGURE 2

A Compiler System Supporting Multiple Languages and Multiple Targets



"TALK IS
CHEAP.
SHOW ME THE
CODE."
-LINUS TORVALDS

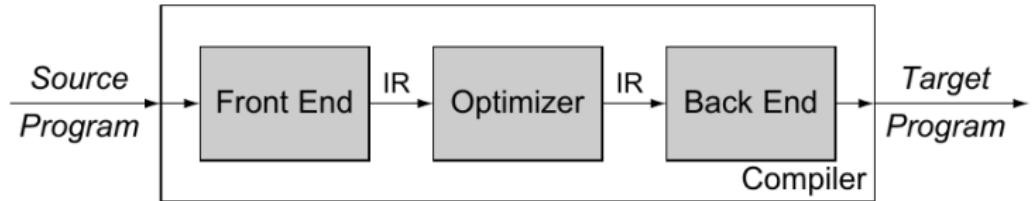
<https://shorturl.at/bGUV> (Clang 的四种读法)

Clang: a C language family frontend for LLVM

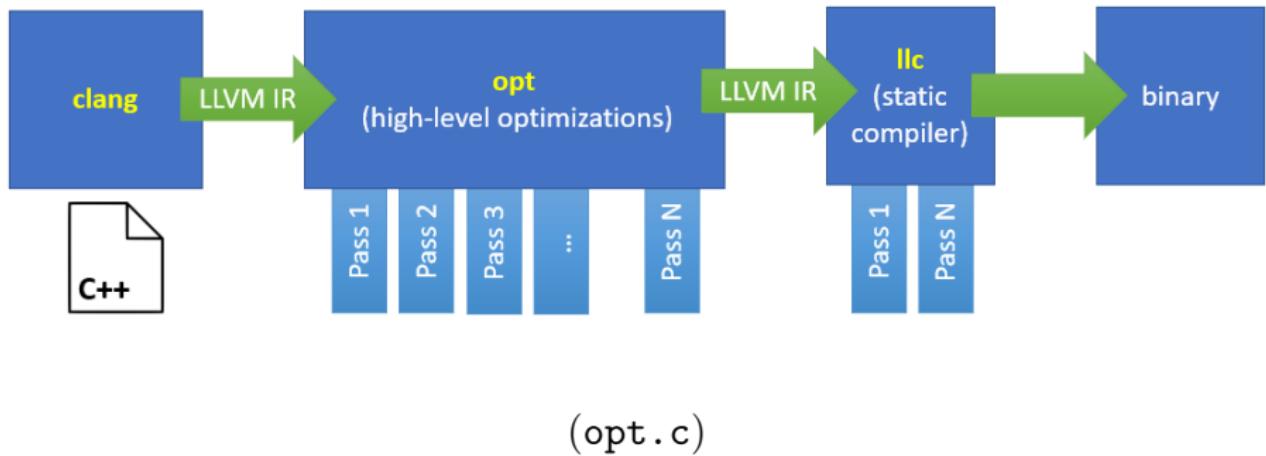
The Clang project provides a language front-end and tooling infrastructure for languages in the C language family (C, C++, Objective C/C++, OpenCL, CUDA, and RenderScript) for the [LLVM](#) project. Both a GCC-compatible compiler driver (`clang`) and an MSVC-compatible compiler driver (`clang-cl.exe`) are provided. You can [get and build](#) the source today.

<https://clang.llvm.org/>

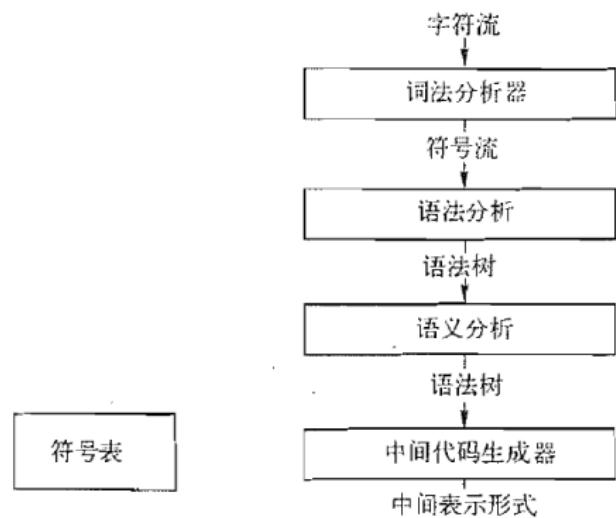
(`factorial.c`)



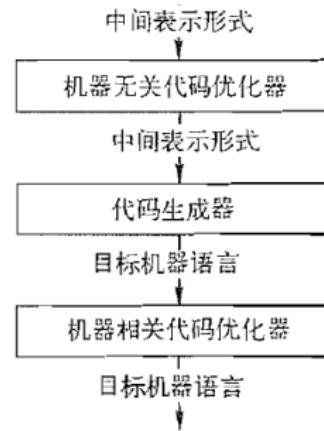
机器无关的中间表示优化



编译器前端

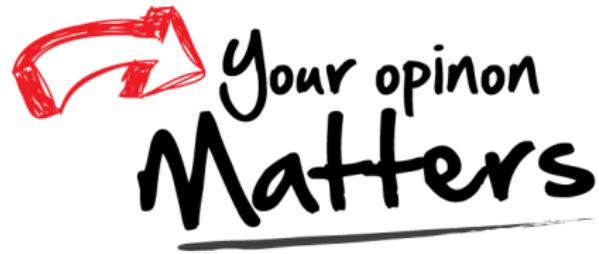


编译器后端



(naming.c)

Thank You!



Office 926

hfwei@nju.edu.cn