

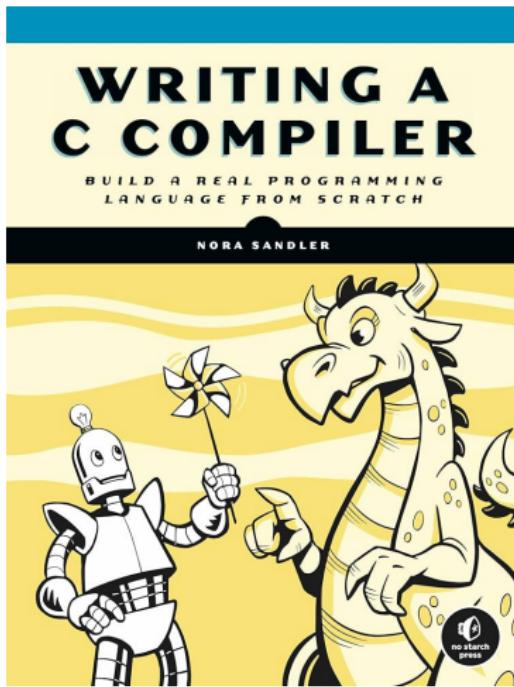
hfwei@nju.edu.cn

20240301 ()

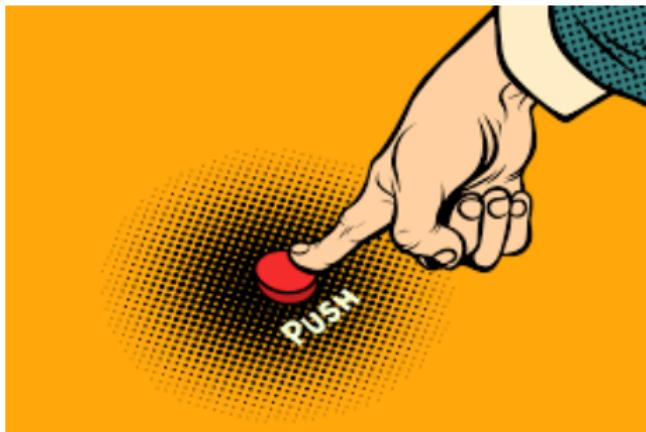


? ?



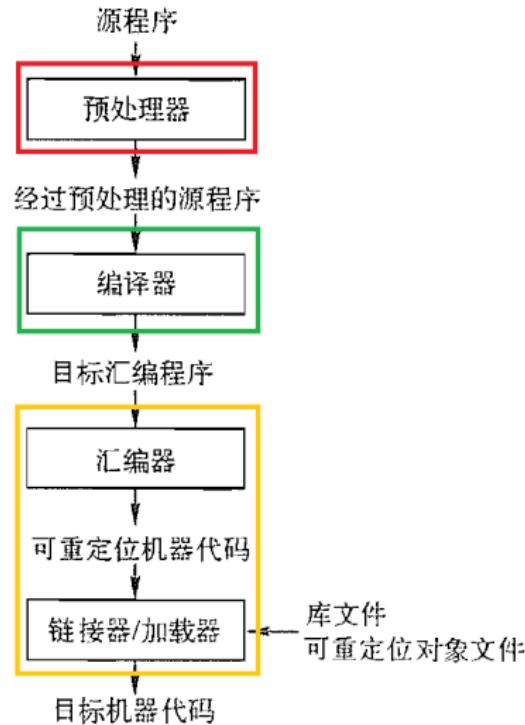


:

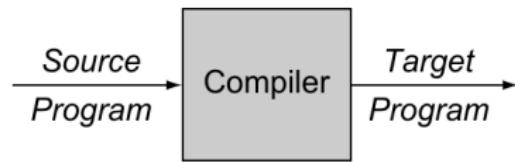


```
gcc factorial.c -o factorial
```

“，？”



“”  $\Rightarrow$  () “” (, )



“”  $\Rightarrow$  () “” (, )



## GopherJS - A compiler from Go to JavaScript

[godoc](#) [reference](#)

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.

“? ?”



(Reduced Instruction Set  
Computer)

“? ?”

<https://godbolt.org/>



(Reduced Instruction Set  
Computer)



<https://godbolt.org/z/6f3o51oeh>  
(factorial.c)

Join Us at RISC-V Summit Europe | June 24-28



Languages

Tech Meetings

Community Meetings

Working Groups Portal

Join



About RISC-V

Membership

RISC-V Exchange

Technical

News & Events

Community

Careers & Learning



# Upcoming Event

## RISC-V Summit Europe 2024

The RISC-V Summit Europe is the premier event that connects the European movers and shakers - from industry, government, research, academia and ecosystem support - that are building the future of innovation on RISC-V. Join us in Munich on June 24-28, 2024.

LEARN MORE



## Join RISC-V International



RISC-V International is a forum for members to connect, build relationships, and share knowledge. It is open to anyone interested in RISC-V.

<https://riscv.org/>

```
book-covers $ cd .. > book-covers.s
1 .file "book-cover.s"
2 .option nopic
3 .attribute arch, "+rv32iipf"
4 .attribute unaligned.access, 0
5 .attribute stack_align, 16
6 .text
7 .align 2
8 compute_the_answer_to_the_ultimate_question_of_life_the_universe_and_everything:
9    .text
10   .align 2
11   .ret
12   .global do_something_1000_times
13   .type do_something_1000_times, @function
14 do_something_1000_times:
15   .addi $a0,$a0,-16
16   .sw $a0,16($sp)
17   .lw $a0,16($sp)
18   .li $a0,1000
19 .Lhi
20   .addi $a0,$a0,-1
21   .call do_something
22   .bne $a0,$a0,-16
23   .lw $a0,16($sp)
24   .lw $a0,16($sp)
25   .addi $a0,$a0,16
26   .j do_something
27 .Lend:
28   .section .rodata.str1.4,"a@P",@progbits,1
29   .align 2
30 .LC0:
31   .ascii "There are 10 types of people in this world "
32   .ascii "those who understand binary and those who don't"
33   .align 2
34 .LC1:
35   .string "Assembly language you must learn!"
36   .align 2
37 .LC2:
38   .ascii "The Unicamp CS course was created in 2009 - "
39   .ascii "The first one in Brazil!"
40 .LC3:
41   .byte 78, 385, 98, 181, 33, 32, 99, 151, 157, 33, 187
42   .byte 159, 113, 119, 32, 65, 89, 97, 73, 73, 33, 87
```

## An Introduction to Assembly Programming with RISC-V

<https://riscv-programming.org/book.html>

```
book-cover.s
Users / edison / Desktop > book-cover.s
1 .file "book-cover.c"
2 .section .text
3 .attribute arch, "+r+32lzipD"
4 .attribute aligned_access, 8
5 .attribute stack_align, 16
6 .text
7 .align 2
8 compute_the_answer_to_the_ultimate_question_of_life_the_universe_and_everything:
9 li $t0, 42
10 add $t0, $t0, $t0
11 .align 2
12 .globl _do_something_3000_times
13 _do_something_3000_times:
14 .type _do_something_3000_times, @function
15 do_something_3000_times:
16    addi $t0,$v0,-16
17    ne $v0,$v0,$sp
18    sw $t0,%1($sp)
19    lw $t0,%1($sp)
20    addi $t0,$v0,16
21    call _do_something
22    bne $v0,$zero,_do_something_3000_times
23    lw $t0,%1($sp)
24    addi $t0,$v0,16
25    addi $t0,$v0,16
26    jr $t0,%1($sp)
27
28 .section .rodata.str1.4, "+W@rrobits,l"
29 .align 2
30 .LC0:
31 .ascii "There are 10 types of people in this world "
32 .ascii "those who understand binary and those who don't"
33 .align 2
34 .LC1:
35 .string "Assembly language you must learn!"
36 .align 2
37 .LC2:
38 .ascii "The Unicamp CS course was created in 1969 = "
39 .ascii "The first one in Brazil"
40 .LC3:
41 .byte 78, 181, 99, 181, 33, 32, 98, 111, 137, 32, 187
42 .LC4:
43 .byte 156, 211, 159, 32, 85, 83, 67, 73, 73, 33, 8
```

# An Introduction to Assembly Programming with RISC-V



<https://riscv-programming.org/book.html>

<http://www.riscvbook.com/>

# RISC-V ,

The screenshot shows a GitHub repository page for 'rars'. At the top, there's a banner message from GitHub stating: "We are having a problem billing the courses-at-nju-by-hfwei organization. Please [update your payment method](#) or call your payment provider for details on why the transaction failed. If the features included in GitHub Free meet your needs, you can downgrade by going to your [Billing settings](#). You can always [contact support](#) with any questions." Below the banner, the repository details are shown: 'rars' (Public), 'master' branch, 3 branches, 12 tags, 365 commits, 25 watchers, 201 forks, 1.1k stars, and 1k issues. The repository description is "RARS -- RISC-V Assembler and Runtime Simulator". The code navigation bar includes 'Go to file', 'Add file', and 'Code'. The commit history lists several commits from various authors, including 'privat' and 'TheThirdOne'. The commits cover topics like fixing text colors, updating pseudo-op tests, and refactoring the Mars package. On the right side, there are sections for 'About', 'Readme', 'View license', 'Activity', 'Stars', 'Watching', 'Forks', and 'Report repository'. A 'Releases' section shows 12 releases, with the latest being 'Lots of small improvements'.

<https://github.com/TheThirdOne/rars>

$Q : ?$

$Q : ?$

(P1 ~ P9): <https://www.bilibili.com/video/BV1EW411u7th>  
( 40 Crash Course Computer Science)

*Q : ?*

(P1 ~ P9): <https://www.bilibili.com/video/BV1EW411u7th>  
( 40 Crash Course Computer Science)



*Q : ?*

(P1 ~ P9): <https://www.bilibili.com/video/BV1EW411u7th>

( 40 Crash Course Computer Science)



*Q : ?*

(P1 ~ P9): <https://www.bilibili.com/video/BV1EW411u7th>

( 40 Crash Course Computer Science)



“, ?”



.fun



install tutorial cheat sheet docs community

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.

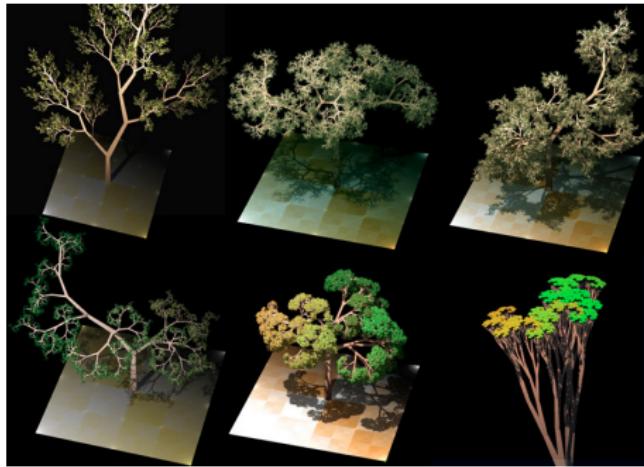
alda repl

alda play -f xxx.alda

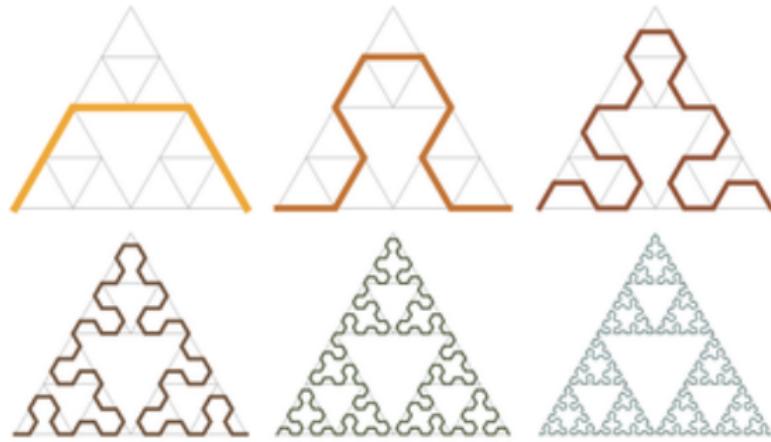
alda @



10 Demo: Alda @ youtube



## Fractal Grower (Play with It!)



Sierpinski Triangle @ wiki



**USEFUL**

- ▶ (.properties)
- ▶ CSV (Comma-Separated Values)
- ▶ JSON (JavaScript Object Notation)

- ▶ (.properties)
- ▶ CSV (Comma-Separated Values)
- ▶ JSON (JavaScript Object Notation)
- ▶ SQL (Structured Query Language)
- ▶ TLA<sup>+</sup>/TLAPS (TPaxos.tla)
- ▶ (Java)
- ▶ C/C++

- ▶ (.properties)
- ▶ CSV (Comma-Separated Values)
- ▶ JSON (JavaScript Object Notation)
- ▶ SQL (Structured Query Language)
- ▶ TLA<sup>+</sup>/TLAPS (TPaxos.tla)
- ▶ (Java)
- ▶ C/C++
- ▶ ( $\text{\LaTeX}$ )
- ▶ (TikZ, Dot/Graphviz)
- ▶ L-System (Cantor Set)

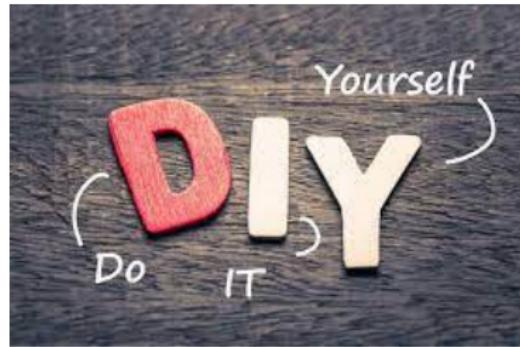
“, , , ?”



: “, ”



紙上得來終覺淺  
絕知此事要躬行



(0%): ,

(0%): ,

(0%):  $\approx 10$ ,  $\leq 3$

(0%): ,

(0%):  $\approx 10$ ,  $\leq 3$

(60%):  $8 \sim 10$

(0%): ,

(0%):  $\approx 10$  ,  $\leq 3$

(60%):  $8 \sim 10$

(40%): ; 3 ;

(0%): ,

(0%):  $\approx 10$  ,  $\leq 3$

(60%):  $8 \sim 10$

(40%): ; 2 ;

23 : 55



: 8G928EBJ

SysY



LO: 16:00

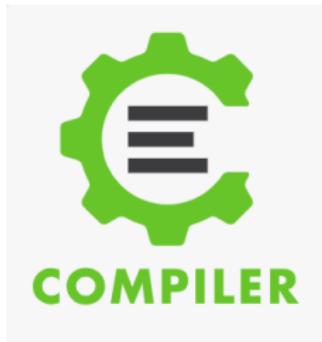


: 0

QQ : 869910463



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



🔍 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

, ...  
<https://oj.compilers.cpl.icu/>

Online Judge

系统导航

- [系统主页](#)
- [作业列表](#)
- [用户设置](#)

---

系统管理

- [用户管理](#)
- [作业管理](#)
- [提交管理](#)
- [备份查询](#)
- [成绩查询](#)

LOGO

2024 南京大学春季学期编译原理校外OJ账号申请 ✎

姓名 \*

请输入您的姓名

学号 \*

请输入您的学号

申请时间 \*

请选择您申请OJ的日期时间

邮箱 \*

请输入您的邮箱。我们会将您的账号密码等信息通过邮箱的形式转发给您

# Zulip

The screenshot shows the Zulip web interface. On the left is a sidebar with navigation links: 'ZULIP' (with a blue icon), 'MESSAGES' (Recent conversations, Inbox, All messages, Mentions, Shared messages, Drafts), 'DIRECT MESSAGES' (Welcome Bot), and 'STREAMS' (core-team, 1-lever-and-dr, general, hxc-general-topics, hxc0, hxc1, hxc2, hxc3, hxc4, hxc5, hxc6, hxc7, hxc8, lab0, lab1, lab2, lab3, lab4, lab5, lab6, lab7, lab8). The main area displays a 'stream events' feed for the '#lecture-0-overview' stream. A notification for 'Notification Bot' is shown, created by Hengfeng Wei, with the description 'Discussion about Lecture 0-overview'. The right side of the interface lists 'USERS' (Hengfeng Wei, 胡海峰, Heze Li, jehewu, mitch, qianpingyi, Qmeket SH, stray0010, The Galaxy) and a link to 'Invite more users'. At the bottom, there are buttons for 'Start new conversation' and 'New direct message', along with a system status bar showing battery level, signal strength, and time (11:31, 2024/03/01).



<https://2024-compilers-at-software-nju.zulipchat.com/join/wxwq3fib56ltlff2mk6qyrz5/>

overview.pdf

overview-handout.pdf

compilers-lectures / 2024 /

Add file ...



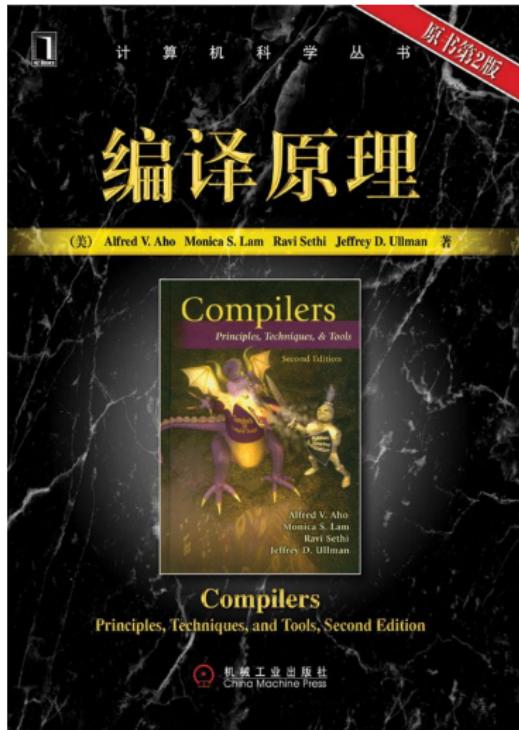
hengxin +material/

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>



“”



Flex:



Bison:



(Since 1988)



Terence Parr (University of San Francisco)

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

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

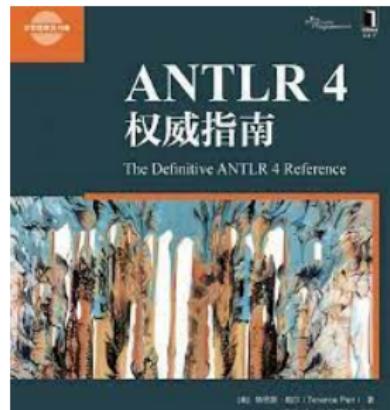
<http://lab.antlr.org/> (Online lab)

The Definitive  
**ANTLR 4**  
Reference



Terence Parr

The  
Pragmatic  
Programmers



ANTLR 4,

The  
Pragmatic  
Programmers

## Language Implementation Patterns

Create Your Own Domain-Specific and General Programming Languages

Edited by Kenneth D. North

Terence Parr



The  
Pragmatic  
Programmers

Language  
Implementation Patterns

## 编程语言 实现模式

Create Your Own Domain-Specific  
and General Programming Languages

[译] Terence Parr 著  
李京生 译  
高鹏鸣 审校



ANTLR 3, ANTLR 4

ANTLR 4

# 自制编译器

How to Develop a Compiler

[日] 青木峰郎 / 著 严圣造 逸云 / 萍

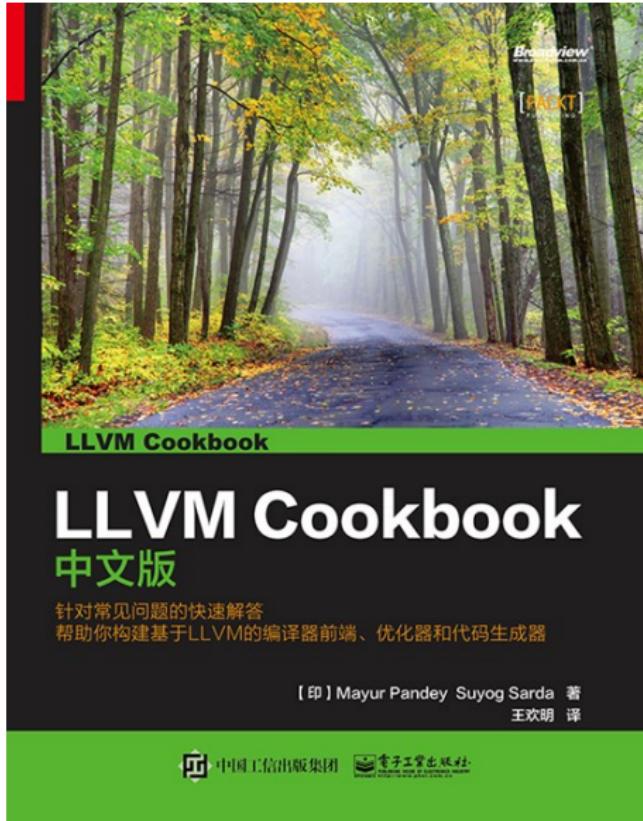
Turing  
图灵精英  
设计丛书

从零开始制作真正的编译器

贯穿编译、汇编、链接、加载的全过程！  
比“教书”更具实践性！

中国工信出版集团 人民邮电出版社  
POSTS & TELECOM PRESS

“ ”



, LLVM (<https://llvm.org/>)



LLVM @ Bilibili





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

## 资源汇总

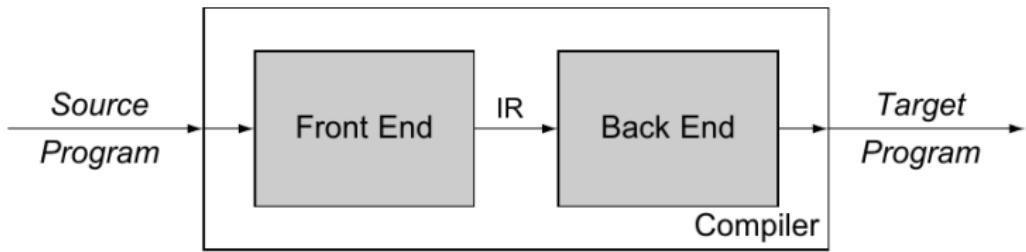
- [compilers-resources @ github](#): 编译原理相关资源
  - [books: 电子书籍](#)

LET'S GET  
STARTED





## IR: Intermediate Representation ()

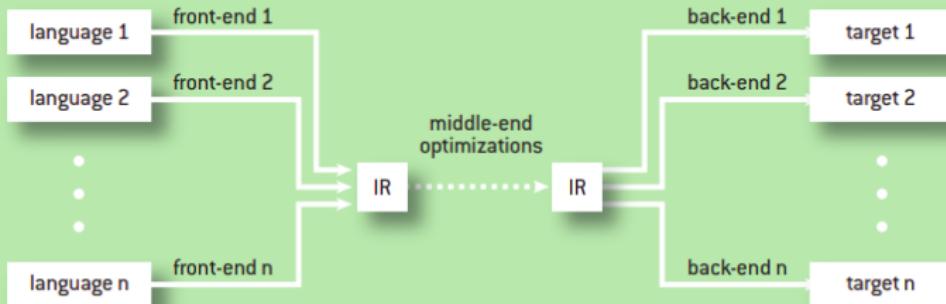


():,

():,

# 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 )

## 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/>

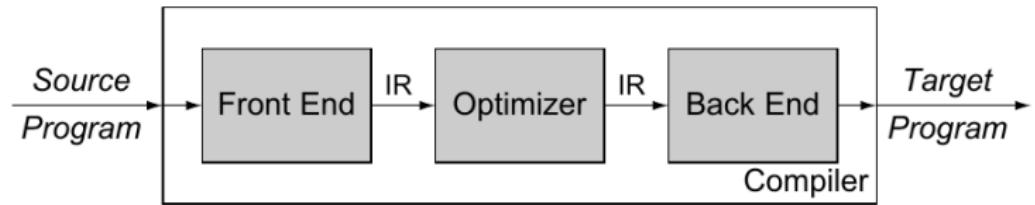
<https://shorturl.at/bGUV> (Clang )

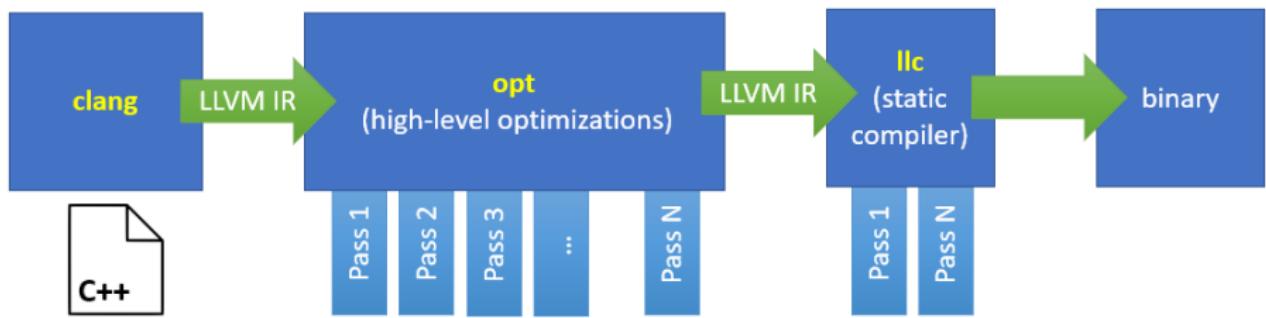
## Clang: a C language family frontend for LLVM

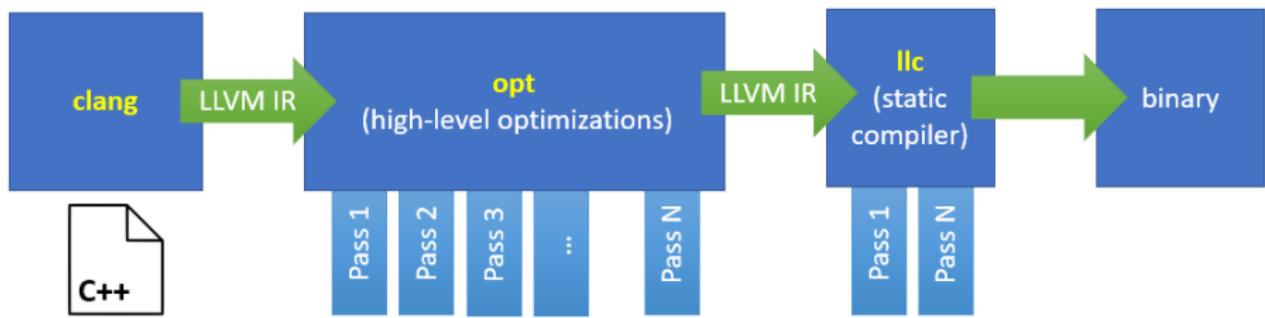
The Clang project provides a language front-end and tooling infrastructure for languages in the C language family (C, C++, Objective C/C++, OpenCL, CUDA, and RenderScript) for the [LLVM](#) project. Both a GCC-compatible compiler driver (`clang`) and an MSVC-compatible compiler driver (`clang-cl.exe`)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

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

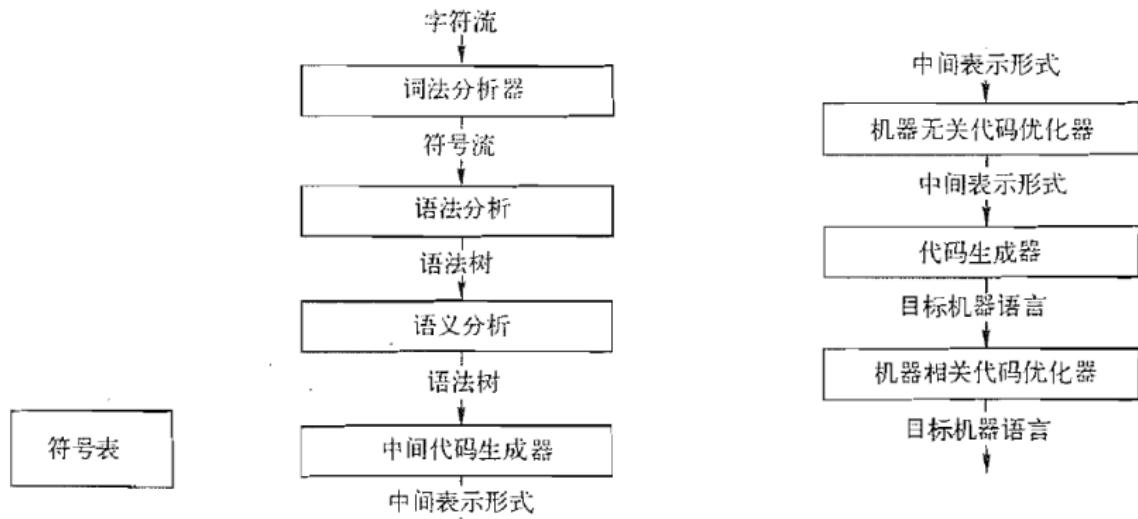
(factorial.c)

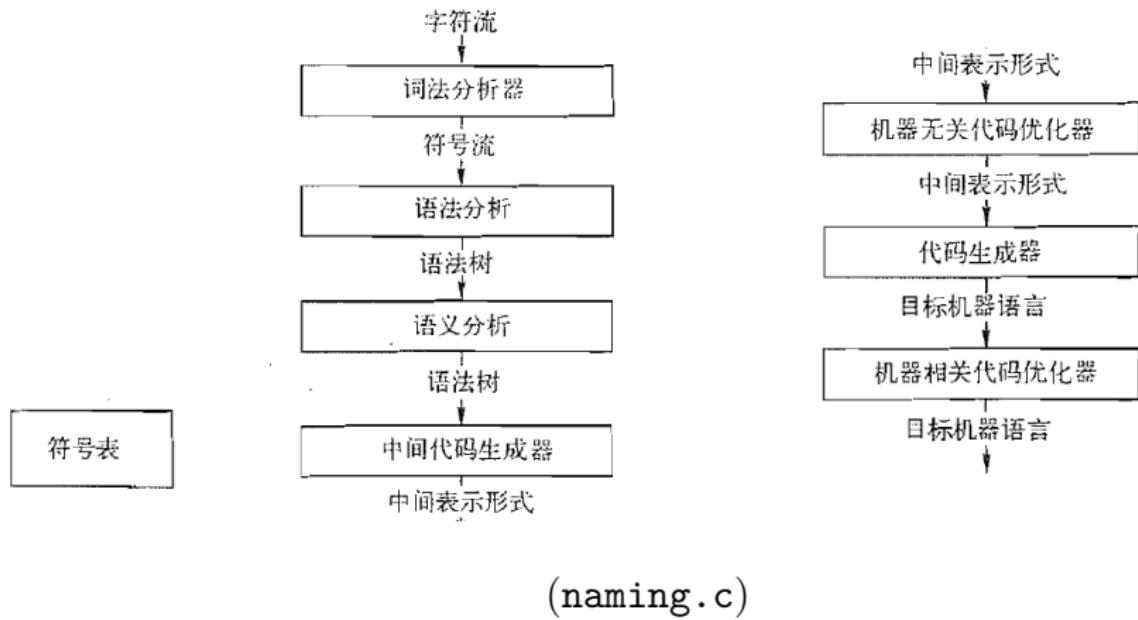




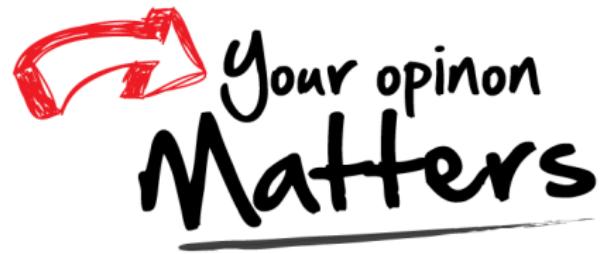


(opt.c)





# Thank You!



Office 926

hfwei@nju.edu.cn