

编译方法有哪些应用?

- 编译器的研究为我们提供计算机科学中的思维方式
 - 理论指导实践
 - 系统的程序设计
 - 从不同的角度看问题
- 更好地理解程序设计语言的本质
- 编译器是所有程序的样本
 - 计算机模拟系统
 - 输出的格式化（文本、网页浏览器）
 - 搜索引擎
 - ...

学习内容和目标

学习内容和目标

- 学习内容

1. (源) 语言的定义、描述方法: 形式语言
2. 编译器的算法: 系统的算法设计
 \Leftarrow 以源程序的分析算法为主
3. 分析算法的实现: 编译器实现相关的技术和实践
4. 源程序到目标程序的等价转换

学习内容和目标

- 学习内容

1. (源) 语言的定义、描述方法: 形式语言
2. 编译器的算法: 系统的算法设计
 \Leftarrow 以源程序的分析算法为主
3. 分析算法的实现: 编译器实现相关的技术和实践
4. 源程序到目标程序的等价转换

- 目标

1. 学习编译的原理、技术和编程实践
2. 系统的编程技术和语言设计的思想
3. 加深对程序设计语言的了解
4. 学会从不同的角度看问题的思想

- 问题 1.3: 概述本课程的学习内容和目标

- 前导课程

1. C 语言：指针，结构体，文件
2. 数据结构：线性表；树（建树、遍历）；哈希表（查找、插入、删除，分离链表）；文件
3. 汇编语言，计算机体系结构，操作系统

- 后继课程

1. 高级程序设计
2. 软件工程
3. 形式语言与自动机
4. 自然语言理解

上机实践与源码研读

为了能够掌握编译的思想、原理和技术

采用理论与实践相结合的学习方式

上机实践与源码研读

为了能够掌握编译的思想、原理和技术

采用理论与实践相结合的学习方式

注重思考方法，贯穿一条主线、补充旁支

一般方法 \Rightarrow 具体实例 \Rightarrow 实践 \Rightarrow 研读源代码



上机实践与源码研读

为了能够掌握编译的思想、原理和技术

采用理论与实践相结合的学习方式

注重思考方法，贯穿一条主线、补充旁支

一般方法 \Rightarrow 具体实例 \Rightarrow 实践 \Rightarrow 研读源代码

- 具体实例：TINY 语言及其编译器

\Leftarrow 深入学习编译器的结构、实现



上机实践与源码研读

为了能够掌握编译的思想、原理和技术

采用理论与实践相结合的学习方式

注重思考方法，贯穿一条主线、补充旁支

一般方法 \Rightarrow 具体实例 \Rightarrow 实践 \Rightarrow 研读源代码

- 具体实例：TINY 语言及其编译器
 \Leftarrow 深入学习编译器的结构、实现
- 实践：通过上机实验，体验编译器的结构、完成完整编译器
 - 编译器项目语言：C-Minus 语言的编译器
 - 认真阅读、理解 TINY 语言的编译器

上机实践与源码研读

为了能够掌握编译的思想、原理和技术

采用理论与实践相结合的学习方式

注重思考方法，贯穿一条主线、补充旁支

一般方法 \Rightarrow 具体实例 \Rightarrow 实践 \Rightarrow 研读源代码

- 具体实例：TINY 语言及其编译器
 \Leftarrow 深入学习编译器的结构、实现
- 实践：通过上机实验，体验编译器的结构、完成完整编译器
 - 编译器项目语言：C-Minus 语言的编译器
 - 认真阅读、理解 TINY 语言的编译器



上机实践与源码研读

为了能够掌握编译的思想、原理和技术

采用理论与实践相结合的学习方式

注重思考方法，贯穿一条主线、补充旁支

一般方法 \Rightarrow 具体实例 \Rightarrow 实践 \Rightarrow 研读源代码

- 具体实例：TINY 语言及其编译器
 \Leftarrow 深入学习编译器的结构、实现
- 实践：通过上机实验，体验编译器的结构、完成完整编译器
 - 编译器项目语言：C-Minus 语言的编译器
 - 认真阅读、理解 TINY 语言的编译器
 - 实践题：与实践相关的思考题
 - 准备题：与实践相关的上机练习题



上机实践与源码研读

为了能够掌握编译的思想、原理和技术

采用理论与实践相结合的学习方式

注重思考方法，贯穿一条主线、补充旁支

一般方法 \implies 具体实例 \implies 实践 \implies 研读源代码

- 具体实例：TINY 语言及其编译器
 \longleftarrow 深入学习编译器的结构、实现
- 实践：通过上机实验，体验编译器的结构、完成完整编译器
 - 编译器项目语言：C-Minus 语言的编译器
 - 认真阅读、理解 TINY 语言的编译器
 - 实践题：与实践相关的思考题
 - 准备题：与实践相关的上机练习题
- 源码研读：在互联网寻找感兴趣的开源编译器源代码，组成学习小组，研读、分析源代码



课程要求

- 认真听课、预习、复习
- 认真完成讲稿上的各类题目和作业
- 认真阅读并理解 TINY 编译器的源代码
- 认真研读开源编译器源代码
- 掌握 TINY 编译器和相关程序的创建和运行
- 成绩评定 (暂定)
 - 平时、(期中、) 项目、研读: 60%
 - 期末: 40%
- 教材和参考书
 - 编译原理及实践, Kenneth C. Loudon 著, 冯博琴冯岚等译, 机械工业出版社
 - 自己动手写 Python 虚拟机, 海纳编著, 北航出版社
 - 形式语言与自动机理论, 蒋宗礼等, 清华出版社
- 课程 qq 群: 680472516

第 1 章 概论

1.1 为什么要用（学）编译器

1.1 为什么要用（学）编译器

- 使用高级语言编程需要编译器

1.1 为什么要用（学）编译器

- 使用高级语言编程需要编译器
- 经过 60 年的研究
大量的（通用）理论和技术
理论与实践的完美结合
广泛的应用

1.1 为什么要用（学）编译器

- 使用高级语言编程需要编译器
- 经过 60 年的研究
大量的（通用）理论和技术
理论与实践的完美结合
广泛的应用
- 编译器是典型的复杂程序

1.2 与编译器相关的程序

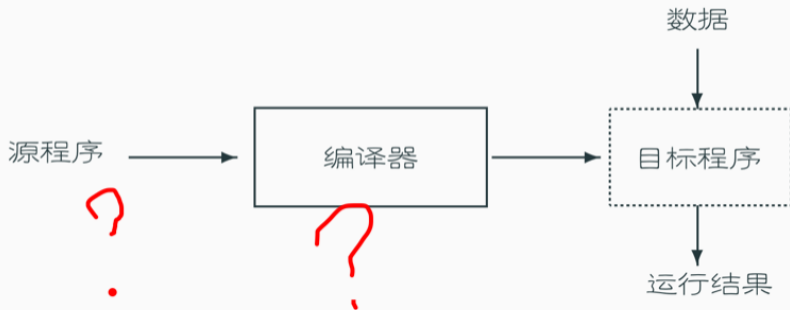
1. 解释程序 (interpreter)
2. 汇编程序 (assembler)
3. 连接程序 (linker)
4. 装入程序 (loader)
5. 预处理器 (preprocessor)
6. 编辑器 (editor)
7. 调试程序 (debugger)
8. 描述器 (profiler)
9. 项目管理程序 (project manager)

利用编译器的程序运行

利用编译器的程序运行



利用编译器的程序运行

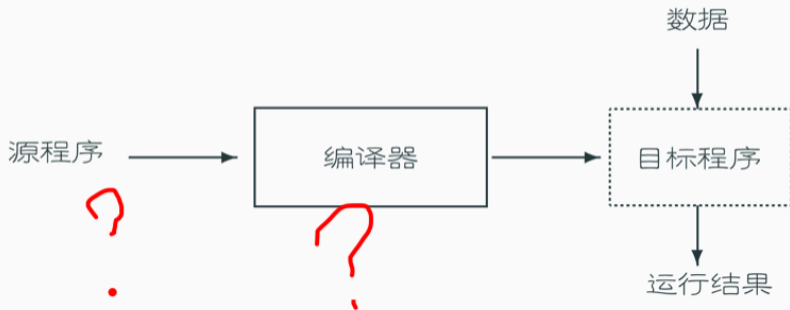


利用编译器的程序运行

?



利用编译器的程序运行



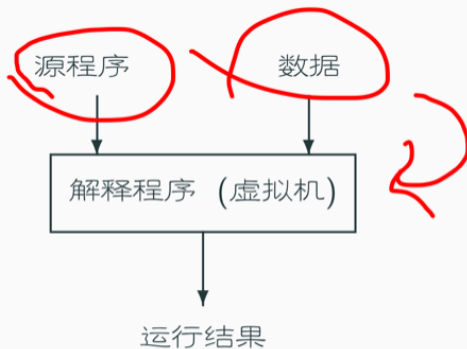
利用编译器的程序运行

?

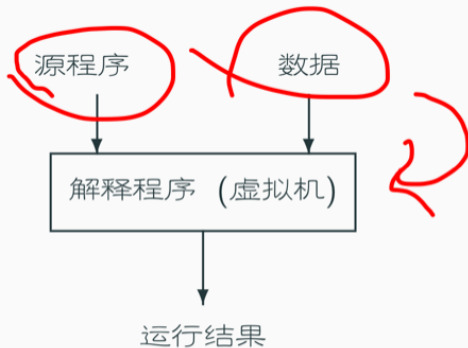


利用解释程序的程序运行

利用解释程序的程序运行



利用解释程序的程序运行



问题 1.4: 说明编译器和解释程序的运行

- VC++
- BorlandC++
- 翻译程序
- 打包的 ppt 文档
- ...

- Python 虚拟机
- 文本编辑器
- 网络浏览器
- 游戏程序
- ...

- ppt,word,pdf 阅读程序
- vb
- TINY、C-Minus 语言的运行环境
- windows 程序 (大部分)
- **思考题**：考虑你常用的软件，说明它们的类型，并解释你的分类