

编译原理中的科学精神与家国情怀

——培养德才兼备的计算机人才

课程思政融入编译原理的意义

- 落实立德树人根本任务，培养德才兼备的计算机人才。
- 避免“重技术、轻价值”，引导学生树立正确科技伦理观。
- 科学精神：严谨性、逻辑性、创新性、普适性
- 家国情怀：科技报国、自主创新、文化自信

编译原理中的思政元素

- **严谨的科学态度：** 编译原理的各个阶段（如词法分析、语法分析、中间代码生成、代码优化等）都要求严格遵循规则和逻辑，任何规则的偏差都可能导致程序运行异常。通过课程学习，有助于培养学生一丝不苟的科学精神，帮助学生在未来的学习和工作中保持高度的责任感和专业性。
- **创新精神与优化意识：** 编译原理中涉及许多算法和优化技术，学生在设计实现时，需要考虑如何提高代码的效率和性能。这种创新精神能够激发学生的创造力，培养他们在遵守规则的基础上勇于探索和突破的能力。

编译原理中的思政元素

- **团队协作与分工：** 编译器的开发通常需要团队协作完成，每个成员都需要对自己的工作和团队整体目标负责。有助于培养学生的沟通能力和团队精神。
- **质量意识与责任担当：** 编译器的质量直接影响到软件的正确性和可靠性。通过学习，让学生认识到他们所学的知识和技术将服务于社会，要确保软件的质量和安全性，培养学生的社会责任感和职业道德。

编译原理中的思政元素

- **系统观念与整体意识：** 编译器是一个复杂的系统，各个阶段相互关联。学生需要从整体上把握编译器的工作原理，理解各个部分之间的关系，有助于培养学生统筹全局、化繁为简的能力，帮助学生在未来的学习和工作中更好地处理复杂问题。
- **前瞻性思维：** 编译器各阶段密切关联，如语法分析的LL(1)、SLR、LR(1)、LALR文法都需要根据规则表（预测分析表或LR分析表）向后看一个token才能决定下一步的动作。需要学生充分考虑任何一个可能的输入，具备一定的前瞻性思维能力，在今后工作过程中养成做好应急预案的习惯，保障工作的正常开展。

经典案例1 从编译器到“中国芯”

- 编译器是芯片开发的关键工具（如GCC、LLVM）
- 在面临国外技术封锁的背景下，龙芯团队凭借坚定的信念和不懈的努力，研发出自主指令系统架构LoongArch，成功实现了从指令集到芯片设计的完全自主化，构建了完整的自主生态体系，为国产芯片的发展树立了标杆。

经典案例2 技术伦理与社会责任

- 技术伦理：编译器生成的代码可能存在安全漏洞，编译器的安全性和可靠性直接影响到软件的运行效果。
- 2018年，编译器领域发生了多起漏洞事件，这些漏洞引发了安全问题
 - Go语言编译器漏洞（CVE-2018-6574）
 - VirtualBox VM逃逸漏洞（CVE-2018-2844）
 - V8 JIT引擎漏洞（CVE-2018-17463）
 - LibSaas越界读取漏洞（CNVD-2018-15185）
- 编译器作为软件开发的基础工具，其安全性直接影响到生成代码的安全性和可靠性。开发者和编译器团队需要密切关注潜在的安全问题，并及时发布补丁以修复漏洞。

Reflections on Trusting Trust

To what extent should one trust a statement that a program is free of Trojan horses? Perhaps it is more important to trust the people who wrote the software.

Ken Thompson在1983年的图灵奖报告（**Reflections on Trusting Trust**）中以一个简单的C编译器为例，指出：源于基础软件的安全问题难以避免。

KEN THOMPSON

panies that employ people like me.) No amount of source-level verification or scrutiny will protect you from using untrusted code. In demonstrating the possibility of this kind of attack, I picked on the C compiler. I could have picked on any program-handling program such as an assembler, a loader, or even hardware microcode. As the level of program gets lower, these bugs will be harder and harder to detect. A well-installed microcode bug will be almost impossible to detect.

构建自主可控、兼容开放的基础软硬件生态体系是实现信息产业健康发展的保障

❖ 自主可控（Independence and Controllability）

由自己主导设计、研制、生产、维护，关键技术不受制于他人，能够保证产品的安全性、使用的可靠性、供货的稳定性。

❖ 基础软硬件生态体系

CPU、操作系统、编译器、数据库、网络（软硬件）

构建自主可控、兼容开放的基础软硬件生态体系是实现信息产业健康发展的保障

❖ 错误认识1：美国对中国的禁运，始于特朗普时代

长期以来，以美国为代表的一些西方国家，奉行和实施对中国的货物、服务和技术出口管制和禁运措施，极大损害了我国的国家利益、产业利益和广大消费者的权益。特别是当我国科技人员自主研发出被禁运装备、技术及软件等以后，美国和其他国家往往立即解禁该类产品与技术，向我国大肆倾销。

节选自：**2015年3月** 《中国计算机学会关于制定反禁运法的建议》

延伸阅读：对中国禁运的那些先进设备和技术（2017-03-18）

https://www.sohu.com/a/129273893_132567

构建自主可控、兼容开放的基础软硬件生态体系 是实现信息产业健康发展的保障

❖ 从巴黎统筹到芯片法案

- 巴黎统筹委员会（COCOM，美国主导）：1949-1994
中国委员会成立于1952年，协调对中国进行最彻底的技术封锁
- 瓦森纳协定（美国主导）：1996至今
中国系首批4个管制国家之一
- 沃尔夫条款（美国单独，全球实施）：2011至今
禁止中美之间的航天合作
- 芯片与科学法案（美国单独，全球实施）：2022年8月至今
限制在中国投资28纳米以下制程的芯片技术
- 美国商务部、联邦通信委员会、国防部、总统行政令黑名单
制裁中国信息产业相关的大多数龙头企业和部分高校

构建自主可控、兼容开放的基础软硬件生态体系 是实现信息产业健康发展的保障

❖ 错误认识2：禁运着力芯片，是因为中国硬件更弱

延伸阅读

top 20 software companies <https://ictbuz.com/software-companies/>

top 10 software companies
<https://www.investopedia.com/articles/personal-finance/121714/worlds-top-10-software-companies.asp>

构建自主可控、兼容开放的基础软硬件生态体系 是实现信息产业健康发展的保障

❖ 错误认识3：禁运着力芯片，是因为软件无法制裁

延伸阅读：

我国高校（哈尔滨工业大学等）被美国禁用Matlab，一场“战争”悄然开始

<https://www.163.com/dy/article/FFI7F06A0545B3M0.html>

Microsoft suspends new sales in Russia

<https://blogs.microsoft.com/on-the-issues/2022/03/04/microsoft-suspends-russia-sales-ukraine-conflict/>

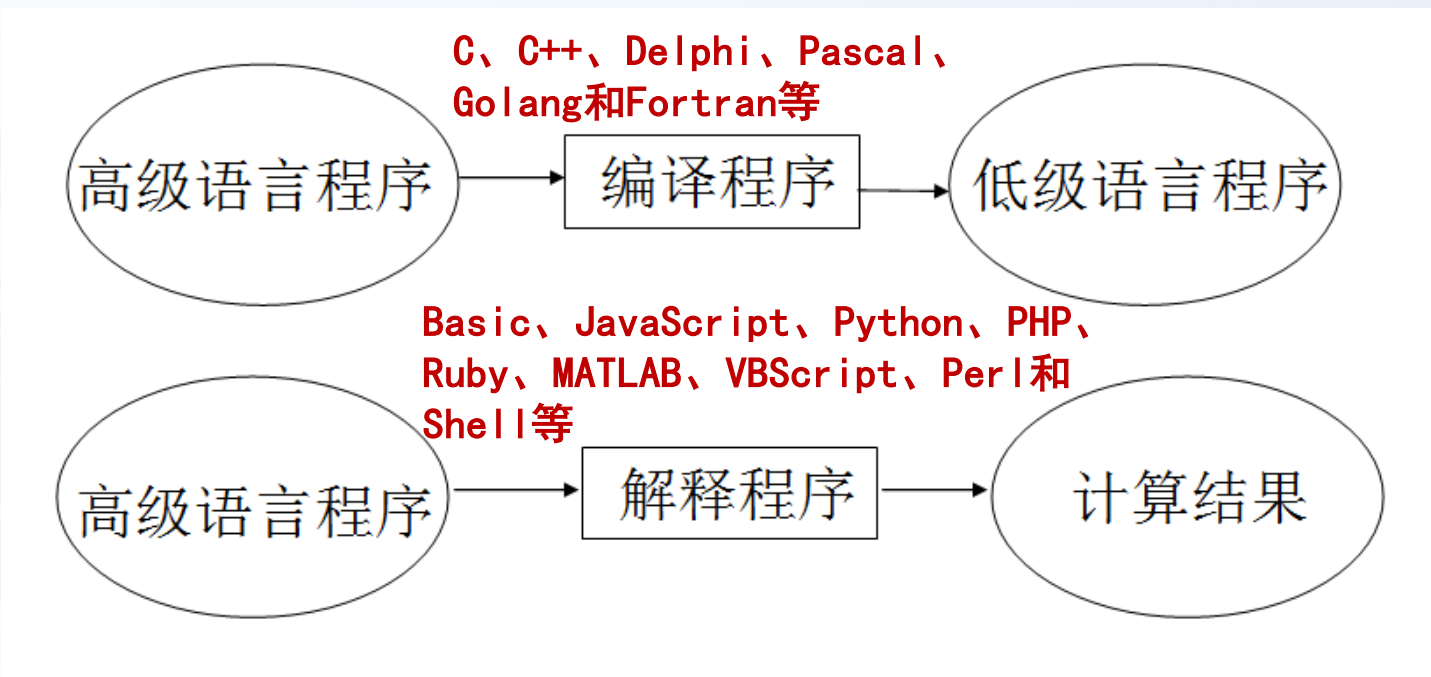
中国计算机学会发布关于IEEE通信学会不当行为的声明

https://page.om.qq.com/page/OMsXgx2p8eG7fG684WyB_bMg0

用唯物主义辩证法思想学习《编译原理》

- **编译执行**：指先将高级语言程序翻译成机器代码，然后，再直接执行与源程序对应的机器指令序列；
- **解释执行**：一边翻译高级语言的语义，一边根据语义产生执行结果；

这两种执行方式都广泛的存在于计算机世界中



用唯物主义辩证法思想学习《编译原理》

- 编译程序好？ **VS** 执行程序好？
- C/C++也大把人用，Python也大把人用，孰优孰劣？

大学生该怎么看待问题、分析问题？

马克思主义辩证法（**唯物辩证法**）是个好工具

- 唯物辩证法的基本规律有三条，即**对立统一规律（矛盾的规律）**、质量互变规律 和 否定之否定规律。
 - 【对立统一规律】唯物辩证法指出：一切存在的事物都由既相互对立、又**相互统一的一对矛盾**组合而成（例如，有光就有影，有正就有负，有得就有失等等）。
 - **矛盾着的双方既对立又统一，从而推动着事物的发展**。因此对立统一规律揭示了事物发展的源泉和动力。马克思和恩格斯的对立统一思想来自黑格尔。

■ 编译程序（代表语言：C、C++）

优点：

1. 执行效率高：编译后的程序执行速度比解释器快；
2. 错误检查方便：编译器通常会在编译阶段捕获许多类型错误，这有助于早期发现和修复问题；

缺点：

1. 开发迭代慢：必须等到编译器完成对代码的编译才能执行，需要多个步骤；
2. 灵活性差：编译后的程序无法直接修改，如果发现了错误或需要修改程序，必须回到源代码中并重新编译；

■ 解释程序（代表语言：Python、PHP）

优点：

1. 快速迭代：由于无需编译过程，可以快速开发和修改程序
2. 灵活性高：直接在解释器中执行代码，可以交互式地开发和测试程序
3. 跨平台：更容易适应不同平台和操作系统

缺点：

1. 执行效率低：由于解释器需要在运行时逐行解释并执行代码，因此其性能通常不如编译后的程序
2. 错误检查能力有限：解释器通常只在运行时检查类型错误和其他错误，这可能导致程序在运行时崩溃
3. 代码安全性问题：由于解释器直接读取并执行源代码，因此可能存在安全风险，例如执行恶意代码

用唯物主义辩证法思想学习《编译原理》

总结一下，事务都有两面性：

- ① 没有轻轻松松的成就： 慵懒没成绩 成绩不慵懒
- ② 没有工作量就没有质量： 学习++ = 娱乐--

学习是这样

工作是这样

人生也是这样

我国保护计算机知识产权法律

计算机软件研发中的知识产权保护：

《著作权法》 《计算机软件保护条例》 《中华人民共和国专利法》

延伸阅读：

出版图书

《计算机软件的知识产权保护》是2009年03月知识产权出版社出版的图书，作者是应明、孙彦。

《计算机软件的知识产权保护》从理论和实务两方面介绍和讨论了计算机软件的知识产权保护问题。

。