

河海大学“编译原理”教学大纲（2020 版）

一、课程名称

编译原理

Principles of Compilers

二、学分学时

3 学分 / 48 学时（48 学时（线下教学及研讨））

三、使用主体教材和线上资源

1. 主体教材

《编译原理》（第 3 版），王生原等，第 3 版，清华大学出版社，2015 年 6 月

2. 线上资源

编译原理与技术，陈意云，<https://www.bilibili.com/video/av18542492/>

四、课程属性

专业主干课

必修

五、教学对象

计算机科学与技术、智能科学与技术、软件工程专业本科生

六、开课单位

计算机与信息学院，物联网工程学院

七、先修课程

离散数学，程序设计基础，高级程序设计语言，汇编语言，数据结构，计算机组成原理，操作系统

八、教学目标

教学总目标：

本课程是计算机科学与技术等专业本科生的学科主干课程。通过本课程学习，使学生掌握编译程序的一般构造原理、方法与技术，包括形式语言与自动机基础理论、词法分析程序构造技术、各类语法分析方法、静态语义计算方法、语法制导的语义计算方法、中间代码生成技术、符号表构造

方法、运行时存储空间组织方法、代码优化方法、目标代码生成技术和整体编译技术等编译的一般方法和实现技术，从而为软件研究、设计与开发奠定必要的理论与技术基础。

本课程从知识、能力、素质等方面确定教学目标：

知识目标：系统地掌握编译器的一般实现过程、编译各个阶段的实现方法以及一些常用的编译设计技术。

能力目标：通过编译一整套理论、方法和技术的系统性学习，掌握基础的软件理论与软件分析方法，能够应用所学知识分析、解决实际工程问题。

素质目标：通过案例分析和小组研讨等方式培养团队协作能力，培养学生良好的职业道德，能够在复杂软件系统分析和设计中理解并遵守计算机专业职业道德和规范，并履行责任。

本课程支撑毕业要求中的 1.3 和 2.3 指标点。

教学分目标：

教学目标 1：

系统掌握词法分析方法、递归下降分析、LL 分析与 LR 分析等常用的语法分析方法、语法制导的语义分析方法、中间代码生成、代码优化、目标代码生成等高级语言编译器构造的一般方法和技术，针对复杂工程问题，综合运用上述专业知识，进行系统性的分析与设计，提出切实可行的解决方案，以实现问题求解。（支撑毕业要求 1.3）

教学目标 2：

掌握形式语言的基本理论，包括文法与语言、推导与规约、语法树、文法与语言二义性、文法分类等；掌握正则文法、正则式与有限自动机理论与方法，了解下推自动机、图灵机等概念；掌握属性文法的基本理论及其计算方法；掌握软件分析的一般方法；能够运用上述理论与方法对计算机科学与技术领域的复杂工程问题进行建模、推演与分析，以获得有效结论。（支撑毕业要求 2.3）

九、课程目标与教学内容和教学环节对应关系表

序号	课程目标	教学内容	教学环节 ^[注1]			
			课堂教学	研讨	实验	上机
1	课程目标1	第一章 编译概论 第三章 词法分析（构造方法部分） 第四章 自顶向下语法分析方法 第五章 LR分析 第七章 静态语义分析和中间代码生成 第八章 代码优化和目标代码生成（优化与生成方法部分）	√	√	√	√
2	课程目标2	第二章 文法和语言 第三章 词法分析（理论部分） 第六章 语法制导的语义计算 第八章 代码优化和目标代码生成（理论部分）	√	√	√	√

注 1：教学环节“实验”与“上机”均在与本课程配套的实践课程《编译原理课程设计》（1 学分）中实施。

十、课程教学内容和基本要求

学生将学习以下课程内容并应达到如下基本要求：

第一章 编译概论

1. 教学内容

编译的基本概念；编译技术产生与发展历程；编译程序及相关概念；编译的一般过程；编译程序的结构与生成过程。

2. 知识要点

编译的概念；编译过程；编译程序结构；编译程序生成过程。

3. 重点难点

编译的概念；编译程序结构；编译程序生成过程。

4. 基本要求

理解并掌握编译的基本概念、编译程序的作用、编译程序的结构以及编译程序的生成过程。

5. 教学方法

课堂授课，课后复习，平时作业。

第二章 文法和语言

1. 教学内容

文法的概念；符号与符号串；文法和语言的形式定义；文法的类型；上下文无关文法及其语法树；句型分析。

2. 知识要点

文法的概念；文法与语言；推导与规约；句子与句型的概念；语法树；文法与语言的二义性；文法分类；自顶向下分析方法；自底向上分析方法。

3. 重点难点

文法的概念；文法与语言；推导与规约；句子与句型；语法树；文法与语言的二义性；文法分类。

4. 基本要求

理解并掌握形式文法和语言的基本理论与方法，能够将形式语言理论用于描述自然语言、计算机语言及语言文本中的各类模式。

5. 教学方法

课堂授课，课后复习，平时作业。

第三章 词法分析

1. 教学内容

词法分析程序的设计；单词的形式化描述工具；有穷自动机；正规式和有穷自动机的等价性；正规文法和有穷自动机的等价性；词法分析程序的自动构造工具。

2. 知识要点

正规式 (RE); 正规文法 (RG); RE 与 RG 之间的等价性; 确定的有穷自动机 (DFA); 不确定的有穷自动机 (NFA); NFA 到 DFA 的等价转换; DFA 化简; RE 与 FA 之间的等价性; RG 与 FA 之间的等价性; 词法分析程序设计。

3. 重点难点

正规式 (RE); 正规文法 (RG); 确定的有穷自动机 (DFA); 不确定的有穷自动机 (NFA); RE 与 FA 之间的等价性; NFA 到 DFA 的等价转换; DFA 化简; 词法分析程序设计。

4. 基本要求

掌握自动机的基本理论与方法, 能够将自动机理论用于解决与系统设计与分析相关的复杂工程问题。

5. 教学方法

课堂授课, 课后复习, 平时作业, 实验操作 (注: 本教学大纲中所有“实验操作”教学模式均在与本课程配套的实践课程中实施——下同)。

第四章 自顶向下语法分析方法

1. 教学内容

自顶向下分析思想; LL(1)文法的判别; 一些非 LL(1)文法到 LL(1)的等价变换; LL(1)分析的实现; LL(1)分析中的出错处理。

2. 知识要点

自顶向下分析思想; 提取公因子; 消除左递归; 递归下降子程序分析; LL(1)分析方法; 应急恢复方法。

3. 重点难点

自顶向下分析; 消除左递归; LL(1)分析方法; 应急恢复方法。

4. 基本要求

掌握自顶向下的 LL(1)分析方法, 包括 LL(1)的判定、转换、分析表构造、分析过程及应急恢复。

5. 教学方法

课堂授课, 课后复习, 平时作业, 实验操作。

第五章 LR 分析

1. 教学内容

LR 分析概述; LR(0)分析; SLR(1)分析; LR(1)分析; LALR(1)分析; 二义性文法在 LR 分析中的应用。

2. 知识要点

LR 分析概述; 可归前缀与活前缀; 识别活前缀的有限自动机; LR(0)项目集规范族构造; SLR(1)分析; LR(1)分析; LALR(1)分析; 二义性文法在 LR 分析中的应用。

3. 重点难点

可归前缀与活前缀的概念; 识别活前缀的有限自动机; LR(0)项目集规范族构造; LR(1)分析; LALR(1)分析。

4. 基本要求

理解自底向上分析的基本原理, 包括活前缀、句柄等关键概念, 掌握四种 LR 分析方法。

5. 教学方法

课堂授课，课后复习，平时作业，实验操作。

第六章 语法制导的语义计算

1. 教学内容

属性文法理论；S 属性文法及其语义计算方法；L 属性文法及其语义计算方法。

2. 知识要点

属性文法理论；S 属性文法及其自下而上分析方法；L 属性文法、L 属性文法的自下而上分析方法、L 属性文法的自上而下分析方法。

3. 重点难点

属性文法理论；S 属性文法及其自下而上分析方法；L 属性文法、L 属性文法的自下而上分析方法、L 属性文法的自上而下分析方法。

4. 基本要求

理解与掌握属性文法理论、S 属性文法、L 属性文法及其语义分析方法。

5. 教学方法

课堂授课，课后复习，平时作业，实验操作。

第七章 静态语义分析和中间代码生成

1. 教学内容

符号表；静态语义分析；中间代码生成。

2. 知识要点

符号表的作用；符号的常见属性及作用；静态语义分析的任务；类型检查；中间代码形式；简单赋值语句、布尔表达式、控制结构（if 语句、while 语句）、数组的翻译。

3. 重点难点

符号表的作用；符号的常见属性及作用；静态语义分析的任务；中间代码形式；布尔表达式、控制结构（if 语句、while 语句）、数组的翻译。

4. 基本要求

理解常见的中间代码表示形式、静态语义分析的任务，掌握赋值语句、布尔表达式、控制结构及数组的翻译方法。

5. 教学方法

课堂授课，课后复习，平时作业、实验操作。

第八章 代码优化与目标代码生成

1. 教学内容

代码优化的概念；基本块、流图、循环；窥孔优化、局部优化；目标代码生成技术。

2. 知识要点

代码优化的概念；基本块、流图、循环；窥孔优化、局部优化；目标代码生成过程。

3. 重点难点

代码优化的概念；窥孔优化、局部优化；目标代码生成过程。

4. 基本要求

理解代码优化和目标代码生成的概念，掌握常见的代码优化方法及一般的目标代码生成过程。

5. 教学方法

课堂授课，课后复习，平时作业，实验操作。

十一、课程思政设计

1. 在相关知识点的教学过程中通过对国、内外编译技术发展、成熟编译产品性能等作对比分析，帮助学生了解中国编译技术的发展历史、现状以及与国外先进技术之间的差距，让学生感悟到我国计算机专业人员肩负着赶超国际先进水平的责任感与使命感。

2. 通过编译原理与方法和编译器设计等方面的讲授与实践，培养学生良好的职业道德，能够在软件系统分析和设计中理解并遵守工程职业道德和规范。

3. 通过案例分析和小组研讨和实践等方式培养学生的团队协作能力。

十二、课程学时安排

本课程为课堂理论教学，围绕编译方法的基本理论以及编译程序构成、编译各阶段的实现方法与技术进行授课。

建议学时分配如下表：（注：实验与上机的学时参见与本课程配套的实践课程的教学大纲）

序号	教学内容	课堂授课（线下）	研讨	实验	上机	总计
1	编译概论	2				2
2	文法和语言	6				6
3	词法分析	8				8
4	自顶向下语法分析方法	6				6
5	LR 分析	8				8
6	语法制导的语义计算	8				8
7	静态语义分析与中间代码生成	6				6
8	代码优化与目标代码生成	4				4
合计		48				48

十三、考核方式

“编译原理”课程为必修课，课程考核方式包括：课堂学习讨论及课后反馈表现（30%左右）+课程考试（70%左右）。

1. 过程性考核（30%左右）

主要考核学生对知识点的理解和掌握程度，其成绩评定由课堂表现、课后作业完成情况、课程参与三部分组成，每部分各占三分之一左右。

2. 结果性考核（70%左右）

考试（闭卷）：主要考核形式语言与自动机理论、编译一般过程、词法分析方法、两大类语法分析方法、语法制导的语义计算方法、静态语义计算方法、中间代码生成方法、符号表构造、代码优化方法、目标代码生成技术等。卷面成绩按 70%左右计入课程总成绩。

课程目标与课程考核环节关系：

序号	课程目标	支撑毕业要求	占比
1	课程目标1	支撑毕业要求1.3	60%
2	课程目标2	支撑毕业要求2.3	40%
总计			100%

十四、教学参考

1. 参考教材

- Alfred V. Aho, Ravi Sethi, Jefferey D. Ullman. Compilers: Principles, Techniques, and Tools. Pearson Education, 2002.
- 赵建华等译，编译器：原理、技术与工具（第2版），机械工业出版社，2009年。（Alfred V. Aho, Ravi Sethi, Jefferey D. Ullman. Compilers: Principles, Techniques, and Tools）
- 陈火旺、刘春林等，《程序设计语言编译原理》第3版，国防工业出版社，2000年。
- 陈意云、张昱，编译原理（第3版），高等教育出版社，2014年。

2. 网络资源

- 中国大学 MOOC 平台上编译原理国家精品在线课程，哈尔滨工业大学开设：
编译原理：<http://www.icourse163.org/course/HIT-1002123007>
- 中国大学 MOOC 平台上编译原理在线课程，国防科技大学开设：
编译原理：<http://www.icourse163.org/course/NUDT-1003101005>
- 网络公开课教学视频，中国科技大学：
编译原理与技术，陈意云：<https://www.bilibili.com/video/av18542492/>

大纲撰写人：邹阳（计信院）、吴云燕（物联网院）

大纲撰写时间：2022.02