《编译原理课程设计》课程教学大纲

（2019版，2022修订）

课程编号：**4043013**

课程名称：**编译原理课程设计**

课程英文名称：**Designing of Compilers**

学时/学分数：**36/1**

课程性质：**必修**

课程类型：**专业限选课**  
适用专业：**计算机科学与技术专业**  
先修课程：程序设计基础，数据结构，编译原理

一、 课程简介

《编译原理课程设计》是为计算机科学与技术专业的专业主干课《编译原理》配套开设的实验课程。通过本课程的上机实践，学生在实验题目的引导下逐步地构造出一个演示性质的微型编译系统，一方面可以加深对《编译原理》课程讲授内容的理解与掌握，另一方面可以加深对高级程序设计语言的理解，并进一步地强化使用高级程序设计语言来表述问题、分析问题和解决问题的能力。

二、 课程教学目标

通过本课程的理论教学和实验训练，使学生具备下列能力：

课程目标1：以现有的高级程序设计语言或其子集为蓝本，改造后作为源语言，或者自行设计源语言，以低级语言形式的伪代码作为目标，设计一个具有一定独特性或原创性的编译器前端。

课程目标2：设计和实现词法分析器、语法分析器和代码翻译器，满足实验题目中的功能性要求，并能够明确指出其缺陷或不足之处以及进一步改进的方向。

课程目标3：能够清晰明确地讲解每一个实验题目的分析思路、设计意图和选用的技术路线，能够现场演示实验成果，能够将实验过程和成果整理形成实验报告。

**表1.课程目标与毕业要求指标点的对应关系**

|  |  |  |  |
| --- | --- | --- | --- |
| **序号** | **毕业要求** | **毕业要求指标点** | **课程目标** |
| 1 | 3.设计/开发解决方案 | 3.3：能够完成信息系统中软硬件的总体设计，并体现创新意识 | 课程目标1：以现有的高级程序设计语言或其子集为蓝本，改造后作为源语言，或者自行设计源语言，以低级语言形式的伪代码作为目标，设计一个具有一定独特性或原创性的编译器前端。 |
| 2 | 5.使用现代工具 | 5.4：针对具体的工程问题或特定的约束条件，选用、移植、改造或开发满足特定需求的软硬件工具，模拟和预测专业问题，并能够分析其局限性 | 课程目标2：设计和实现词法分析器、语法分析器和代码翻译器，满足实验题目中的功能性要求，并能够明确指出其缺陷或不足之处以及进一步改进的方向。 |
| 3 | 10.沟通 | 10.1：能就专业问题进行有效的口头和书面表达，包括介绍、解释、质疑、辩论等，理解与业界同行和社会公众交流的差异性 | 课程目标3：能够清晰明确地讲解每一个实验题目的分析思路、设计意图和选用的技术路线，能够现场演示实验成果，能够将实验过程和成果整理形成实验报告。 |

三、 课程内容安排和基本要求

**(一)教学内容、要求及教学方法**

**表2.课程目标与教学内容和方法的对应关系：**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **序号** | **教学内容（知识点）** | **教学要求** | **学时** | **教学方式** | **课程**  **目标** |
| 1 | 确定源语言：选择或设计一个源语言。 | 【掌握】对现有的高级程序设计语言的理解  【了解】不同语言的特征、特性 | 0 | 自学 | 1 |
| 2 | 词法分析：设计、实现一个词法分析器。 | 【重点掌握】：从输入串中识别出各类单词符号；  【掌握】：缓冲区在输入输出中的应用；单词符号的数据结构设计；  【了解】：词法分析阶段的错误处理。 | 8 | 代码研读、分析讨论、上机实践 | 1，2，3 |
| 3 | 语法分析：设计、实现一个语法分析器。 | 【重点掌握】：递归下降语法分析器的构造技术；  【掌握】：自上而下的语法分析方法：思想、存在的问题和解决办法；树结构的存储、构造和展示技术；  【了解】：非递归的预测分析方法、算符优先分析方法和LR分析方法。 | 12 | 代码研读、分析讨论、上机实践 | 1，2，3 |
| 4 | 代码翻译：将设计好语法分析器升级改造为一遍扫描的代码翻译器。 | 【重点掌握】：递归下降翻译器的设计；  【掌握】：常用语句的翻译方法；一遍扫描的语法制导翻译方法；语义子程序的设计；  【了解】：抽象语法树。 | 12 | 代码研读、分析讨论、上机实践 | 1，2，3 |
| 5 | 组装编译器前端：  对前三个实验的工作进行整理，将其成果组装、升级成为一个完整的编译器前端，能够将高级语言形式的源程序翻译成低级语言形式的伪代码。 | 【掌握】：编译器的工作过程；前端和后端；  【了解】：模块化的程序设计。 | 4 | 上机实践 | 1 |

备注： ①第1、4部分视课程进度可安排为完全自学内容，不提交书面成果。

四、课程考核

**（一）考核与评价方式及成绩评定**

本课程考核方式如下：

对于实验教学内容中2、3、4三个部分的实验题目，要求学生在规定时间内完成相关分析、设计和编码工作，并提交实验报告。本课程的考核成绩由三次实验报告的成绩求取平均值得来。

各考核环节所占分值比例可根据具体情况进行调整，建议值及考核细则如下：

**表3.课程目标达成考核与评价方式及成绩评定**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 课程目标 | 毕业要求指标点 | 考核与评价方式及成绩比例（%） | | | 成绩比  例（%） |
| 实验报告1 | 实验报告2 | 实验报告3 |
| 课程目标1 | 支撑毕业要求3.3 | 10 | 10 | 10 | 30 |
| 课程目标2 | 支撑毕业要求5.4 | 20 | 20 | 20 | 60 |
| 课程目标3 | 支撑毕业要求10.1 | 3 | 4 | 3 | 10 |
| 合计 | | 33 | 34 | 33 | 100 |

注：该表格中比例为每一部分评分占课程总成绩的比例。

**2、考核与评价标准**

**表4.实验报告考核与评价标准**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **课程目标** | 评价标准 | | | | |
| 优秀 | 良好 | 中等 | 及格 | 不及格 |
| 90-100 | 80-89 | 70-79 | 60-69 | 0-59 |
| **1** | 能够独立完成源语言和目标语言的设计,逻辑清晰，概念明确，完全满足或高于实验要求；编译器各模块功能明确，接口清晰，高内聚，低耦合；有一定创新性。 | 能够独立完成源语言和目标语言的设计,完全满足实验要求；编译器各模块功能明确，接口清晰；有一定创新性。 | 能够独立完成源语言和目标语言的大部分设计工作,基本满足实验要求；编译器各模块功能明确，接口清晰；有创新性的尝试。 | 能够在老师指导下或其他同学帮助下完成源语言和目标语言的大部分设计,基本满足实验要求；编译器各模块功能明确；有创新性的尝试。 | 不能够完成源语言和目标语言的设计,或设计结果不能满足实验要求；编译器各模块功能不明确，接口不清晰，结构混乱。 |
| **2** | 能够独立完成实验要求的分析和设计工作，能够将设计成果编程实现，且满足或高于实验要求；能够明确指出缺陷或不足之处以及进一步改进的方向。 | 能够独立完成实验要求的分析和设计工作，能够将设计成果编程实现，大致满足实验要求。能够明确指出缺陷或不足之处以及大致的改进的方向。 | 能够在老师指导下或其他同学帮助下完成实验要求的分析和设计工作，能够将设计成果编程实现，大致满足实验要求。能够明确指出缺陷或不足之处。 | 能够在老师指导下或其他同学帮助下完成实验要求的大部分分析和设计工作，能够将设计成果编程实现，程序运行能得到有效结果。 | 不能够完成实验要求的分析和设计工作，不能够将设计成果编程实现，无可运行程序。 |
| **3** | 能够清晰明确地讲解实验题目的分析、设计过程，能够现场演示实验成果，能够将实验过程和成果整理形成较为规范的实验报告。 | 能够较为明确地讲解实验题目的分析、设计过程，能够现场演示实验成果，能够将实验过程和成果整理形成实验报告。 | 能够讲解实验题目的大部分分析、设计过程，能够现场演示实验成果，能够将实验过程和成果整理形成实验报告。 | 能够讲解实验题目的大部分分析、设计过程，能够现场演示实验成果，延时提交实验报告或报告中存在较多疏漏。 | 不能够讲解实验题目的主要分析、设计过程，无法现场演示实验成果，未能按时提交实验报告或报告不完整、错误较多。 |

2019年12月31日制订

2022年2月21日修订