东北大学秦皇岛分校本科课程教学大纲

一、课程基本信息

<table>
<tr><td>开课单位</td><td colspan="3">计算机与通信工程学院</td><td>课程代码</td><td colspan="5">3030113069</td></tr>
<tr><td>中文名称</td><td colspan="8">编译原理</td></tr>
<tr><td>英文名称</td><td colspan="8">Compilation Principle</td></tr>
<tr><td>总学时</td><td>60</td><td rowspan="2">理论学时</td><td rowspan="2">48</td><td rowspan="2">实验学时</td><td rowspan="2">12</td><td rowspan="2">上机学时</td><td rowspan="2">0</td><td rowspan="2">设计学时</td><td rowspan="2">0</td><td rowspan="2">课外学时</td><td rowspan="2">0</td></tr>
<tr><td>学　分</td><td>3.5</td></tr>
<tr><td>课程类型</td><td colspan="8">☐理论课程　☑理论（含实验）课程　☐实践类课程</td></tr>
<tr><td>课程类别</td><td colspan="8">☐通识类　☐学科基础类　☑专业方向类　☐实践类</td></tr>
<tr><td>适用专业</td><td colspan="8">计算机科学与技术</td></tr>
<tr><td>先修课程</td><td colspan="8">数据结构、离散数学、C++程序设计</td></tr>
<tr><td>建议开课学期</td><td colspan="8">☐1-1 ☐1-2 ☐2-1 ☐2-2 ☑3-1 ☐3-2
☐4-1 ☐4-2 ☐5-1 ☐5-2 ☐*（代表每学期都开设）</td></tr>
<tr><td>考核方式</td><td colspan="4">☑考试 ☐考查</td><td>成绩分制</td><td colspan="3">☑百分制 ☐五级制
☐二级制</td></tr>
<tr><td>使用教材</td><td colspan="8">1. 王生原 ，董渊，张素琴，吕映芝，蒋维.《编译原理（第 3 版）》，清华大学出版社，2015.</td></tr>
<tr><td>参考书及教学资源</td><td colspan="8">1. Alfred V. Aho, Monica S. Lam, Ravi Sethi, Jeffrey D. Ullman 著; 赵建华, 郑滔,. 戴新宇译. 编译原理(第 2 版), 机械工业出版社, 2017 年.
2. 张莉, 史晓华, 杨海燕, 金茂忠. 编译技术, 高等教育出版社, 2016 年</td></tr>
<tr><td>其他说明</td><td colspan="8">无</td></tr>
</table>

二、课程简介

中文描述

编译原理是计算机科学与技术专业的一门重要专业基础课程，涉及编译器的设计原理和实现技术，是计算机领域的一个重要分支，与计算机专业的其他许多领域紧密相关，有着广泛的应用和一定的启发与指导作用。

本课程讲授的内容包括传统、经典的编译知识：词法分析、语法分析、语义分析、代码生成等，还包括编译器构造的基本原理和技术以及一些有用的编译器构造工具：Lex、Yacc 等。课程的目的是通过深入浅出的教学，帮助学生理解和掌握计算机程序从一种语言形式到另一种语言形式的等价转换过程，深刻理解高级语言在计算机上运行的机理，掌握计算机软件和硬件之间联通的过程，最终具备将理论知识应用于解决实际问题的能力。

通过本课程的学习，学生将能够建立起坚实的理论知识框架，这对于他们未来在学术上进一步深入研究或在职业生涯中从事复杂系统的工作至关重要。在实践操作方面，学生将会清楚地理解一个编译程序是如何工作的，理解任何一个程序设计语言的实现机制，具有使用编译构造工具开发编译程序的经验，同时将所学的常用技术和算法应用于类似的软件设计和实现中，这将大大增强他们在实际工作中分析和解决复杂系统问题的能力。

更重要的是，本课程还注重培养学生的家国情怀和四个自信。通过学习编译知识，学生不仅能够获得实用的技能，还能够在学习过程中理解编译程序构造对国家“自主可控、安全可靠”基础软件研发的重要性，具备科技报国的家国情怀和使命担当，树立起正确的世界观、人生观和价值观，从而更好地服务于国家的发展和人民的福祉。

English Description

Compilation Principle is an important foundational course in the field of computer science and technology, which involves the design principles and implementation techniques of compilers. It is an important branch of the computer field and is closely related to many other fields of the computer profession. It has a wide range of applications and certain inspiration and guidance.

The content includes traditional and classic compilation knowledge such as lexical analysis, syntax analysis, semantic analysis, code generation, etc. It also includes the basic principles and techniques of compiler construction, as well as some useful compiler construction tools such as Lex and Yacc. The purpose of the course is to help students understand and master the equivalent conversion process of computer programs from one language form to another, deeply understand the mechanism of high-level languages running on computers, master the process of connecting computer software and hardware, and ultimately have the ability to apply theoretical knowledge to solve practical problems.

Through the study of this course, students will be able to establish a solid theoretical knowledge framework, which is crucial for their future in-depth academic research or career work in complex systems. In terms of practical operation, students will have a clear understanding of how a compiled program works, understand the implementation mechanism of any programming language, have experience in using compilation and construction tools to develop compiled programs, and apply commonly used techniques and algorithms learned to similar software design and implementation. This will greatly enhance their ability to analyze and solve complex system problems in practical work.

More importantly, this course focuses on cultivating students' patriotism and four confidences. Through compilation knowledge, students can not only acquire practical skills, but also understand the importance of compiling program construction in the development of "independent, controllable, safe and reliable" basic software for the country during the learning process. They have a sense of patriotism and mission to serve the country through science and technology, establish correct worldviews, outlooks on life, and values, and better serve the development of the country and the well-being of the people.

三、课程目标

1、知识目标：

1-1：掌握形式语言理论与编译实现相关的基础概念，包括编译程序的结构、过程、文法和语言的形式定义、有穷自动机、正规式等方面的基本概念、基本理论；

1-2：掌握编译程序构造的基本原理与技术，包括词法分析、语法分析、属性文法和语法制导的翻译、语义分析、中间代码生成和优化的基本方法，从形式语言理论的角度，进一步认识与理解程序设计语言及其与编译程序的联系；

1-3：掌握编译系统的体系结构和基本工作过程。

2、能力目标：

2-1：掌握编译系统的一般构造原理、基本实现技术和常用自动构造工具，培养编译器的设计和实现的能力；

2-2：能够设计并实现简单语言的词法、语法、语义分析以及目标代码生成功能；

2-3：能够基于编译原理的知识实现机器无关的中间代码和基于目标机的目标代码的优化；

2-4：通过科学思维方法的训练，培养学生运用科学原理解决实际问题的工程能力，最终能为学习本专业后继课程和从事与编译器构造、实现有关的技术工作奠定基础。

3、**德育/素养目标：**

3-1：展示本专业在新时代中国特色社会主义建设中的成就和当前要解决的重大课题；

3-2：具有良好的团队协作意识和能力、较强的表达能力和人际交往能力；

3-3：理解编译系统架构及理论知识对于刻画工程实践问题的重要意义；

3-4：认知当前全球，特别是我国编译系统理论的发展对提升中国工程关键技术及核心竞争力的重要意义；

3-5：认知大国工匠精神的内涵及时代意义，增强专业认同感、民族责任感。

表 1 课程目标与毕业要求的对应关系

毕业要求	指标点	课程目标	关联性（H/M/L）
1、工程知识：能够将数学、自然科学、工程基础和专业知识用于解决计算机相关工程问题。	指标点 1-1：掌握数学、自然科学、工程基础和专业知识，并能够将其运用到复杂计算机科学与工程问题的恰当表述中；	1-1 、 2-4 、 3-1、3-3、3-4	H
	指标点 1-3：掌握计算机组成原理、操作系统与编译原理等基础知识，掌握计算机系统的技术原理；	1-1 、 1-2 、 2-1 、 2-2 、 2-3 、 3-1 、 3-3、 3-4	H
2、问题分析：能够应用数学、自然科学和工程科学的基本原理，识别、表达、并通过文献研究分析计算机工程问题，以获得有效结论。	指标点 2-2：能够有效分析和处理软件工程、编译系统、计算机操作系统，数据库等方面的技术问题；	1-1 、 1-2 、 2-1 、 2-3 、 3-2、3-4、3-5	M

3、设计/开发解决方案：能够设计针对计算机工程问题的解决方案，设计满足特定需求的软硬件系统，并能够在设计环节中体现创新意识，考虑社会、健康、安全、法律、文化以及环境等因素。	指标点 3-1：具备本专业所需的设计/开发技能，能够设计针对复杂计算机工程问题的解决方案，设计满足特定需求的系统；	1-2 、 1-3 、 2-2 、 2-3 、 3-3、3-4	M
4、研究：能够基于科学原理并采用科学方法对复杂工程问题进行研究，包括设计实验、分析与解释数据、并通过信息综合得到合理有效的结论。	指标点 4-1：能够基于科学原理并采用科学方法，在本专业相关理论指导下对复杂工程问题设计实验进行研究；	1-2 、 2-1 、 2-4、3-1	H

四、主要教学内容

（一）理论教学

第 1 章 绪论（支撑课程目标 1-1、1-3、2-2、3-1、3-3、3-4、3-5）

【1】知识点：

（1）什么是编译
（2）编译系统的结构
（3）词法分析概述
（4）语法分析概述
（5）语义分析和中间代码生成概述
（6）目标代码生成概述
（7）代码优化概述
（8）编译程序和解释程序
（9）符号表和出错管理概述

【2】学习目标：

了解： 编译的基本过程和与编译相关的工具及技术。
理解： 语言翻译程序的作用和翻译的方式。
掌握： 编译程序的结构和阶段。

【3】重点内容：

编译器的概念，编译的过程，编译程序的结构：词法分析概述，语法分析概述，语义分析和中间代码生成概述，编译阶段的组合，编译程序和解释程序，符号表和出错管理概述。

【4】难点内容：

编译程序的结构：词法分析概述，语法分析概述，语义分析和中间代码生成概述，符号表和出错管理概述。

【5】思政案例：

在“高级程序设计语言及其编译器的发展历程”的理论教学中，对比我国过去的技术落后和现在的技术飞跃，对比 IBM、微软、惠普等国际知名公司的在主流编译器上的垄断及近年来华为方舟编译器，鸿蒙操作系统的开源为国产编译器的发展开辟了新篇章。强调科学技术是第一生产力，激发学生的责任心、自信心。

第 2 章：程序设计语言及其文法（支撑教学目标1-1、1-2、2-1、3-1、3-3）

【1】知识点：

（1）形式语言基础；

（2）文法的定义；

（3）语言的定义

（4）文法的分类；

（5）CFG 的分析树

【2】学习目标：

了解：形式语言的描述工具。

理解：文法的定义。

掌握：句型推导的方法。

【3】重点内容：

文法和语言的形式定义，文法的类型，句型、句子的定义，推导和规约的概念，句型的推导。

【4】难点内容：

文法和语言的形式定义，推导和规约的概念，句型的推导。

【5】思政案例：

（1）每一门程序设计语言都是同学们用以实现具体功能时所选择的一种程序构造工具，这些工具的背后都有属于程序设计语言的共同的原理抽象和运行机制。编译原理要学习的就是从具体的语言中抽象出程序设计语言所共同遵守的规则，然后利用这些规则让计算机能够自动的构造翻译各种计算机语言的工具，包括编译器，解释器等高级语言翻译工具。这就是从具体到抽象，然后从抽象到具体的过程。通过形式语言及其文法和计算机语言及其编译器的实现之间的关系，让学生理解抽象和具体及两者之间的转化关系，掌握科学思维方法的使用。

（2）运用形式化方法对感性问题进行描述形成抽象的、逻辑严密的数学概念，达到归纳和演绎的统一。通过现代语言学家乔姆斯基生成式文法的提出，论述科学精神的重要性。我们应该坚持理论联系实际，注重培养学生独立思考能力、实践能力、创新能力和批判性思维，以及追求真理、崇尚创新、实事求是的科学精神。乔姆斯基除了作为语言学家也是一个社会批判家和政治活动家，其对于美国社会的认识极为深刻，提出了“根据纽伦堡审判的法律，二战以来的所有美国总统都可以被送上绞刑架。(1990 年)”等重要的语录。

第 3 章：词法分析（支撑教学目标1-1、1-2、1-3、2-1、2-2、3-3）

【1】知识点：

（1）词法分析程序的设计；

（2）正则表达式；

（3）正则定义；

（4）有穷自动机 FA；

（5）有穷自动机的分类；

（6）从正则表达式到有穷自动机；

（7）从 NFA 到 DFA 的转换；

（8）识别单词的 DFA；

（9）词法分析阶段的错误处理；

（10）词法分析程序的自动构造工具。

【2】学习目标：

了解：词法分析程序的功能和构造及其组织，了解词法分析程序的自动构造过程。

理解：文法和有穷自动机在词法分析中的作用和机制，正则式与正则文法以及有穷自动机之间的等价关系。

掌握：NFA 到DFA 的转换算法和DFA 的化简算法。

【3】重点内容：

使用正则定义描述程序设计语言中的单词、NFA 到DFA 的转换、DFA 的化简。

【4】难点内容：

使用正则定义描述程序设计语言中的单词、NFA 到DFA 的转换、DFA 的化简。

第 4 章：自顶向下语法分析（支撑教学目标1-1、1-2、1-3、2-1、2-2、3-3）

【1】知识点：

（1）自顶向下语法分析法概述；
（2）文法转换；
（3）LL（1）文法；
（4）递归的预测分析法；
（5）非递归的预测分析法；
（6）预测分析中的错误处理。

【2】学习目标：

了解：自顶向下语法分析的基本原理和机制，预测分析阶段的错误以及处理方法。
理解：确定的自顶向下语法分析的原理和分析过程。
掌握：文法的转换，LL（1）文法的判别，递归子程序法和非递归的预测分析法。

【3】重点内容：

文法的转换、LL（1）文法的判别、LL（1）分析表的构造、递归的子程序和非递归的预测分析法。

【4】难点内容：

LL（1）文法的判别、LL（1）分析表的构造、非递归的预测分析法。

【5】思政案例：

（1）详细讲解递归和非递归的预测分析法的原理和算法，强调其严谨性和准确性。

（2）鼓励学生思考在人工智能时代背景下，面向 AI 型语言的设计，如何改进预测分析法，提高其效率和准确性。这可以激发学生的创新精神，培养他们敢于挑战传统、勇于探索新方法的勇气。

第 5 章：自底向上语法分析（支撑教学目标1-1、1-2、1-3、2-1、2-2、3-3）

【1】知识点：

（1）自底向上语法分析概述；
（2）LR 类分析概述；
（3）LR（0）分析；
（4）SLR（1）分析；
（5）LR（1）分析；
（6）LALR（1）分析；
（7）二义型文法的 LR 类分析；
（8）LR 类分析中的错误处理；
（9）语法分析的自动构造工具。

【2】学习目标：

了解：自底向上语法分析的基本原理和机制，LR 类分析阶段的错误以及处理方法。
理解：LR 类语法分析的原理和分析过程。
掌握：LR 类自动机的构造，LR 类文法的判别，主要包括 LR（0）、SLR（1）、LR（1）和 LALR（1）文法的判别及其相应自动机的构造以及对应分析表的构造，使用 LR 类自动机分析高级程序设计语言。

【3】重点内容：

LR 类文法的分析思想、LR 类自动机的构造、LR 类分析表的构造以及分析算法的实现。

【4】难点内容：

LR 类自动机的构造、LR 类分析算法的实现。

【5】思政案例：

（1）LR 分析法的关键问题是解决问题理论依据和总体思想的理解和领悟，是后续 LR（0）和 SLR（1）分析法的一般性原则和方法。站在统筹全局和立足整体的角度重新诠释 LR 分析法总体思想，使学生深入理解 LR 分析法分析问题和解决问题的思想和方法。

（2）LR 分析法的核心问题是句柄的确定和最左归约过程实现，最左归约的是当前句型的句柄。辩证思维是通过概念判断、推理等思维形式对符合客观事物辩证发展过程的正确反映。运用归纳与演绎、具体与抽象等辩证思维方法重新推演可归前缀等概念的引入及句柄的确定，使学生深刻领会 LR 分析法的精髓和实质，增强辩证思维能力和透过现象看本质的能力，塑造正确的世界观和价值观。

（3）LR 分析法设计者-计算机图灵奖获得者高纳德倾注几十年心血的《计算机程序设计的艺术》著作，每一版力求随着有关领域的最新进展不断更新内容，至今影响着世界上无数计算机专业工作者。激励学生要有坚忍不拔的态度和持之以恒的决心，以科学思想方法和科学精神面对我国计算机领域的“卡脖子”问题，立志破解难题。

第 6 章：语法制导翻译（支撑教学目标1-3、2-2、3-3）

【1】知识点：

（1）语法制导翻译概述；

（2）语法制导的定义（SDD）；

（3）SDD 的求值顺序；

（4）S-属性定义与 L-属性定义；

（5）语法制导翻译方案（SDT）；

（6）在非递归的预测分析过程中进行翻译；

（7）在递归的预测分析过程中进行翻译；

（8）L-属性定义的自底向上翻译。

【2】学习目标：

了解：语法制导翻译的基本思想和相关概念。

理解：语法制导定义和语法制导翻译方案。

掌握：属性文法的概念，两种属性文法在非递归和递归的预测分析过程中的翻译。以及 L-属性定义的自底向上翻译。

【3】重点内容：

属性文法的概念、语法制导定义、语法制导翻译方案、在自顶向下和自底向上的语法分析过程中的翻译过程。

【4】难点内容：

语法制导翻译方案、在自顶向下和自底向上的语法分析过程中的翻译过程。

【5】思政案例：

（1）语法制导翻译需要对语法规则和语义进行精确的定义和处理，这体现了工匠精神中的精益求精。引导学生在学习和实践中追求卓越，不断提高自己的技术水平。

（2）在实际的软件开发中，语法制导翻译往往是一个团队合作的过程。不同的人可能负责不同的模块，需要相互配合、协调工作。通过小组项目等方式，培养学生的团队协作能力，让他们学会倾听他人意见、发挥各自优势，共同完成任务。

（3）编译技术在软件产业中起着至关重要的作用，而软件的质量和安全性直接关系到用户的利益。引导学生认识到自己作为未来的软件开发者，肩负着社会责任，要确保自己编写的代码安全可靠、符合规范。

第 7 章：语义分析和中间代码生成（支撑教学目标1-3、2-2、2-3、3-3）

【1】知识点：

（1）声明语句的翻译；

（2）赋值语句的翻译；

（3）控制流语句的翻译；

（4）代码回填；

（5）Switch 语句的翻译；

（6）过程调用语句的翻译。

【2】学习目标：

了解：语义分析和中间代码生成的思想。

理解：高级语言程序中声明语句和执行语句的语义分析和中间代码生成过程。

掌握：声明语句、赋值语句、控制流语句和过程调用语句的翻译。

【3】重点内容：

声明语句、赋值语句、控制流语句和过程调用语句的翻译。

【4】难点内容：

含有数组引用的声明语句和赋值语句的翻译，控制流语句的翻译。

【5】思政案例：

中间代码是编译的产物，通过介绍编译执行和解释执行的相同点和不同点，既能明确的认识到两种执行方式各自的优缺点，又能深入的理解在计算机语言领域，编译执行和解释执行长期“两条腿走路”的历史和现状。“两条腿走路”方针是中国共产党根据客观实际，调查分析，正确认识问题、解决问题的一次伟大的成功范例。科技领域看似不谋而合的相似之处，绝非偶然，而是科学的认识论和方法论的必然产物。专业课程上，结合历史事实，通过类比的方式，讲授编译执行和解释执行“两条腿走路”的技术路线，既有利于学生深刻理解专业上的科技内容，同时，也能激发起学生对社会主义宏伟建设事业的共鸣，从而增强中国特色社会主义的道路自信。

第 8 章：运行存储分配（支撑教学目标 1-3、2-2、2-4、3-3）

【1】知识点：

（1）运行时存储组织；

（2）静态存储分配；

（3）栈式存储分配；

（4）调用序列和返回序列；

（5）非局部数据的访问；

（6）堆式存储分配；

（7）符号表的建立。

【2】学习目标：

了解：数据空间的不同使用方法和管理方法。

理解：堆式和栈式动态存储分配，符号表的组织。

掌握：非局部数据访问的方法和实现原理。

【3】重点内容：

栈式存储分配的实现；非局部数据访问。

【4】难点内容：

非局部数据访问。

【5】思政案例：

（1）通过介绍存储分配技术的发展历程，让学生了解科学家们在不断探索和创新中推动计算机技术进步的精神。例如，从早期的简单静态存储分配到现代的复杂动态存储分配，每一次进步都是科学家们勇于创新、不断尝试的结果。鼓励学生在学习过程中敢于提出自己的想法和创新解

决方案，培养他们的创新意识和实践能力。

（2）结合存储分配对资源利用的影响，引导学生思考如何在程序设计中节约内存资源，提高程序的可持续性。例如，合理选择存储分配策略可以减少内存浪费，降低能源消耗。培养学生的社会责任感，让他们认识到自己作为计算机专业人员，在开发软件时应该考虑到资源的合理利用和环境保护。

第 9 章：代码优化（支撑教学目标 1-3、2-2、2-3、3-3）

【1】知识点：

（1）基本块和流图；

（2）常用的代码优化方法；

（3）基本块的优化；

（4）数据流分析。

【2】学习目标：

了解：代码优化的概念和基本技术；

理解：常用局部优化、循环优化和全局优化的方法；

掌握：基本块优化和数据流分析的方法。

【3】重点内容：

常用的代码优化方法；基本块的优化；数据流分析与全局优化。

【4】难点内容：

数据流分析与全局优化。

【5】思政案例：

随着信息技术的飞速发展，软件系统的复杂性和资源消耗不断增加。目标代码优化作为提高程序效率的重要手段，其背后蕴含着对资源合理利用和社会责任的重要考量。引导学生思考，作为未来的软件开发者和工程师，他们的每一次代码优化都是对社会和环境的一份贡献。通过技术手段减少资源浪费和环境污染，是每个技术工作者的社会责任。结合中国传统文化中的“天人合一”和“节约资源”的理念，引导学生认识到绿色计算不仅是技术需求，也是文化自觉和文化自信的表现。再以某大型电商平台的订单处理系统如何通过优化减少服务器负载和能耗。通过本案例，学生不仅掌握编译原理中目标代码优化的基本知识和方法，还能深刻理解绿色计算和可持续发展的重要性，学会在技术优化中融入环保和社会责任的理念，为将来成为具有社会责任感和使命感的优秀工程师奠定基础。同时也增强学生对中国传统文化的自信和对国家政策的认同感。

第 10 章：目标代码生成（支撑教学目标 1-3、2-2、2-3、3-3）

【1】知识点：

（1）代码生成器的主要作用；

（2）一个简单的目标机模型；

（3）指令选择；

（4）寄存器选择。

【2】学习目标：

了解：代码生成的主要作用和基本技术；

理解：简单目标机模型；

掌握：代码生成时的指令选择和寄存器选择的方法。

【3】重点内容：

指令选择和寄存器选择。

【4】难点内容：

指令选择和寄存器选择。

【5】 思政案例：

目标代码生成技术在不断发展和进步，鼓励学生勇于探索新的方法和技术，提高目标代码的质量和效率。介绍一些前沿的研究成果和创新应用，激发学生的创新思维和探索欲望。

（二）实验教学

实验教学主要围绕实现一个简单语言的编译器开展，实验教学的项目名称及学时分配相关信息如表 2 所示。

表 2 实验教学信息表

序号	实验项目名称	实验学时	每组人数	实验类型	实验性质
1	词法分析实验	4	2	设计性	必做
2	语法分析实验	4	2	设计性	必做
3	语义分析和代码生成	4	2	设计性	必做

实验项目一：词法分析实验（支撑教学目标 1-1、1-2、1-3、2-1、2-2、3-1、3-2、3-3、3-4、3-5）

【1】 实验内容：

设计、编写、调试一个具体的词法分析程序，加深对词法分析原理的理解。

【2】 实验目的：

掌握词法分析方法。

【3】教学要求：

根据给定的文法实现一个小型编译器的词法分析阶段要完成的任务。根据给定的文法设计并实现词法分析程序，从源程序中识别出单词，记录其单词类别和单词值，数据结构和与语法分析程序的接口自行定义；类别码需按给定表中格式统一定义。

【4】实验方式：程序设计

【5】实验场地和设备：综合楼 1108/1208，计算机，一体化实验教学平台。

【6】思政案例：

“从编译器的发展历程看中国信创（国产操作系统、国产编译器、国产数据库等）自主产业的生态系统建立”。了解计算机程序设计语言的发展；了解编译系统的发展；掌握组成编译系统的几个阶段和编译程序的逻辑模型；熟悉与编译器有关的程序；掌握词法分析程序的主要任务以及如何词法分析程序的两种构造方法：手工编设计实现、利用自动机自动构造。

实验项目二：语法分析实验（支撑教学目标 1-1、1-2、1-3、2-1、2-2、3-1、3-2）

【1】实验内容：

设计、编写、调试一个具体的语法分析程序，加深对语法分析原理的理解。

【2】实验目的：

理解语法分析原理，掌握语法分析的设计实现方法。

【3】教学要求：

根据给定的文法实现一个小型编译器的语法分析阶段要完成的任务。请根据给定的文法设计并实现语法分析程序（推荐自顶向下的递归子程序分析方法），基于词法分析实验所识别出的单词，

进一步识别出各类语法成分。需按文法规则，用递归子程序法对文法中定义的所有种语法成分进行分析。

【4】实验方式：程序设计

【5】实验场地和设备：综合楼 1108/1208，计算机，一体化实验教学平台。

【6】思政案例：

强调团队合作的必要性，引导学生发挥特长，实验的过程中可以分组、组员分工明确，激发学生认识行业中团队协作的重要性。

实验项目三：语义分析和代码生成实验（支撑教学目标 1-1、1-2、1-3、2-1、2-2、3-1、3-2）

【1】实验内容：

设计、编写、调试一个具体的语义分析和代码生成程序，加深对语义分析和代码生成的理解。

【2】实验目的：

理解语义分析原理和代码生成过程，掌握语义分析和代码生成的设计实现方法。

【3】教学要求：

在词法和语法分析实验的基础上，为编译器实现语义分析和代码生成功能。需根据文法规则及语义约定，采用自顶向下的语法制导翻译技术，进行语义分析并生成目标代码。

【4】实验方式：程序设计

【5】实验场地和设备：综合楼 1108/1208，计算机，一体化实验教学平台。

【6】思政案例：

在实验中，引导学生考虑程序的安全性、可靠性和可维护性等因素，培养社会责任感。例如，要求学生在代码生成过程中遵循良好的编程规范，提高程序的质量和可维护性。

强调职业道德的重要性，要求学生遵守学术规范和知识产权法律法规，不抄袭他人的实验成果。培养学生的诚信意识和职业操守。

五、教学安排与学时分配

本课程以课堂教学为主，辅以实验教学形式。课堂教学围绕本课程的基本概念、基本理论、基本原理和实践操作进行授课。课程围绕立德树人根本任务，以培养德智体美劳全面发展社会主义建设者和接班人为总目标，以编译系统基本架构为基础，逐步展开，深入讲解词法分析、语法分析、语义分析、中间代码生成、代码优化和目标代码生成以及过程中涉及到的实现技术。课程充分利用线下课堂、线上教学资源和思政案例开展教学活动，发挥理论知识传授、实验实践操作及价值引领作用，厚植家国情怀和强国理念，扎实基础知识，提升实践技能和创新精神，树立正确的人生观、世界观和价值观。线下面授课堂的学时分配如表 3 所示。

表 3 学时分配表

讲　　课　　内　　容		学时				
		讲课	实验	上机	设计	课外
第 1 章	绪论	2	0	0	0	0
第 2 章	程序设计语言及其文法	2	0	0	0	0
第 3 章	词法分析	6	4	0	0	0
第 4 章	自顶向下语法分析	6	4	0	0	0

第 5 章	自底向上语法分析	8				
第 6 章	语法制导翻译	6	0	0	0	0
第 7 章	语义分析和中间代码生成	8	4	0	0	0
第 8 章	运行存储分配	4	0	0	0	0
第 9 章	代码优化	4	0	0	0	0
第 10 章	目标代码生成	2	0	0	0	0
小计		48	12			
总计	60					

六、教学方法

1、讲授基本原理，理论联系实际，强调高级语言编译系统的组成及设计原理，培养学生创新能力。

2、采用多媒体课件、网络教学和传统教学相结合的教学方式，赋予课堂生机。

3、理论教学、讨论形式相结合，调动学生的学习积极性，培养学生的自学能力、团队协作能力和解决实际问题的能力，结合国产编译器相关知识及发展历史展示我国高级程序设计语言的发展历程，培养学生责任感并强化学习兴趣。

七、成绩评定

1、本课程考试形式为闭卷，按百分制评定。结合平时作业、课堂提问、课堂交流、期中测验成绩和期末考试成绩综合评定成绩。

（1）平时成绩占 10%，主要包括课堂交流、作业、线上资料学习等综合评定。

（2）实验成绩占 30%，主要为上机实验操作、系统测评和撰写报告。

（3）期末成绩占 60%，主要为试卷成绩（闭卷测试）。

2、所缺实验、作业等超过应完成量 1/3 者，或缺课学时超过计划学时 1/3 者，或实验考核成绩不合格者，取消该课程考核资格。

八、评价及持续改进

1、教学大纲

考核周期：4 年，修订周期：4 年。

改进措施：课程负责人组织课程团队所有教师讨论后提出改进意见，经教研室主任审核后由教学院长批准。

2、成绩评定考核

考核周期：1 学年，评价依据：学生最终成绩综合分析。

改进措施：根据学生最终成绩的分布，进行相应的内容调整和优化，并改进教学方法。

撰写：　吕艳霞

审核：　万聪

批准：　王聪

日期：　2024.07.05