《汇编语言程序设计》课程大纲

一、课程名称：汇编语言程序设计

二、课程性质：必修、理论课

三、学时与学分：24学时，1.5学分

四、课程先导课：计算机概论，C语言程序设计，数字逻辑（第一、二章）

五、课程简介

本课程是计算机科学与技术专业的专业基础课程，是深入理解计算机软硬件协同工作的纽带。主要讲授80X86等CPU的指令系统及宏汇编语言，分析、编写、调试汇编语言程序（包括分支循环程序、子程序、模块化程序、中断处理程序等）的方法。通过汇编语言程序的设计，讨论如何灵活利用CPU、内存和外部端口的资源，有效支撑高级语言等软件完成其功能的原理与方法。它是其他后续课程（如“计算机组成原理”、“接口技术”、“操作系统”、“编译原理”、“逆向工程分析技术”、“信息安全技术”等）的基础。

六、课程目标

通过相关教学活动，帮助学生理解计算机（以基于80X86的计算机为主）基本工作原理，掌握汇编语言程序设计的方法，为进一步深入理解和运用其他计算机软硬件系统打下基础。

课程的具体目标包括：

**目标1：**掌握汇编语言的概念与特点，理解计算机的基本工作原理；掌握阅读和分析汇编语言程序的方法；加深对高级语言内部实现机制以及计算机软、硬件系统的整体化理解；能从汇编语言的角度描述和求解计算机的工程问题。

**目标2：**掌握编写、调试汇编语言程序的方法；能设计出较充分利用了汇编语言优势的软件功能模块。

七、课程目标对毕业要求的支撑关系

|  |  |
| --- | --- |
| **支撑的毕业要求二级指标点** | **对应课程目标** |
| **1.2**能针对计算机复杂工程问题的具体对象进行建模和求解 | 目标1 |
| **3.2**能为计算机复杂工程问题解决方案设计满足特定需求的软/硬件模块 | 目标2 |

八、教学设计及对课程目标的支持

**第一章 计算机基本工作原理**

**1.教学目标**

1）了解计算机系统的基本组成，80X86 等CPU的发展历史、基本结构及特点；

2）掌握CPU执行指令的基本过程；

3）掌握汇编语言的基本概念、数值表示方法、内存编址方法、物理地址形成原理、标志寄存器中标志位的含义等；

4）了解汇编语言程序的基本框架。

本章教学支持的课程目标为目标1。

**2.教学重点**

1）CPU执行指令的基本过程

这是理解计算机基本工作原理及汇编语言程序设计方法的基础。学生要准确把握CPU、内存、机器指令与数据等之间的关系。

2）标志寄存器中标志位的含义

要准确理解与程序设计密切相关的各个标志位的含义，为汇编语言程序的条件控制打下基础。

**3.教学难点**

1）段地址、偏移地址与物理地址的形成

2）进位与溢出标志，有符号数与无符号数相关的标志位

**4.教学环节设计**

围绕教学重点和教学难点，综合应用课堂讲授、讨论、作业、共享资源学习等教学形式。

1）讨论

围绕逻辑地址与物理地址之间的转换方法、有/无符号数的运算特点与标志位的关系等问题展开。

2）作业

围绕数值表示、物理地址计算、加减运算与标志位设置等内容布置。

3）共享资源学习

通过精品资源共享课平台等网络资源，了解CPU指令系统的概况以及相关的课程与知识体系。

**5.思政元素**

在介绍Intel 80X86处理器的发展简史时介绍一下华为的ARM64及其发展史，简单对比一下。CPU的发展史其实就是一部创新发展的历史，速度、能耗、指令集（代码复杂性）、工艺与性价比等是CPU发展中不断需要面临的问题，华为的选择体现了华为的创新和担当精神，值得我们学习和努力。

**第二章 寻址方式**

本章的主要知识点包括寻址方式的概念，6种寻址方式的特点及其运用。

1.**教学目标**

1）熟悉各个寻址方式的作用及使用的格式、限制条件等；

2）正确选用恰当的寻址方式完成指令的操作目的；

3）掌握不同寻址方式之间的转换方法。

本章教学支持的课程目标为目标1。

**2.教学重点**

1）存储器寻址方式

理解变量、寄存器等如何构成相应的存储器寻址方式，各寻址方式与C语言相关变量的关系等。

2）存储器寻址方式之间的转换方法

理解不同存储器寻址方式内在的转换关系，为支持各种特殊要求的程序设计打下基础。

**3.教学难点**

1）寻址方式与操作数之间的关系

理解寻址方式如何算出操作数地址，何时针对的是存储单元的地址，何时针对的是存储单元里的内容。

2）各个存储器寻址方式之间的内在关系

理解各个存储器寻址方式中寄存器、变量等的灵活搭配关系，根据不同的要求，选择合适的搭配。

**4.教学环节设计**

围绕教学重点和教学难点，综合应用讨论、作业等教学形式。

1）讨论

围绕不同寻址方式的应用场景展开。

2）作业

对给定的指令语句中的寻址方式进行分析评判，或根据需要撰写合适的寻址方式。

**5.思政元素**

在讲解指令语句的六种寻址方式时，先举例说明在生活中找一个人的方式有多少种，进而说明生活中人的行为方式，是如何在CPU的设计上体现出来。技术产品是为人的生活工作服务的，也是可以来源于人的生活工作的。另外，不同的寻址方式也说明，做好一件事也不是只有一条路，所以，对不同的意见和做法要有一定的包容，尤其是在合作的时候。从而强调从事技术活动，同样需要培养好的人文素养。

**第三章 宏汇编语言**

本章的主要知识点包括各种汇编语句的构成，变量定义等伪指令语句，机器指令语句的学习方法与常用的机器指令简介，指令的特点与使用技巧。

**1.教学目标**

1）掌握包括段、变量、标号、常量等在内的各种符号的定义方法，理解各种符号的属性；

2）掌握机器指令语句的学习方法，能够自学并使用新的机器指令；

3）能熟练编写正确的伪指令语句和机器指令语句。

本章教学支持的课程目标为目标1。

**2.教学重点**

1）变量的多种初始化方法，数据段在内存中的存放形式，变量的访问；

2）典型的传输指令、加减与乘除法指令、位操作指令等的理解，包括指令功能、格式、标志位影响、约束条件等。

**3.教学难点**

1）变量的地址与访问

以变量的属性为基础，借助数据段在内存中存放的示意图，说明变量地址的首字节地址特性；以一般传输指令为例，说明对变量的访问方法与技巧，能够实现对数据段任意字节地址、任意自定义类型的数据访问。

2）指令的特点及其在程序设计与优化等方面的灵活运用

**4.教学环节设计**

围绕教学重点和教学难点，综合应用讨论、作业、共享资源学习等教学形式。

1）讨论

本章讨论主要围绕变量在内存中的存放与访问展开。

2）作业

本章作业主要围绕变量定义、常见指令语句的编写等布置。

3）共享资源学习

完成本课程精品资源共享课平台相关内容的学习，阅读Intel指令手册。

**5.思政元素**

殊途同归地解决一个问题，如用多种做法给ax寄存器置0，不再有标准答案，更多地需要善于从不同的视角去看问题和解决问题，培养学生的科学素养和创新精神。

**第四章 程序设计的基本方法**

本章的主要知识点包括程序设计方法、分支程序设计、循环程序设计、子程序设计。通过本章的学习，帮助学生理解在高级语言中的各种程序结构是如何在机器底层实现的，以及不同语言之间混合编程时的约束条件。

**1.教学目标**

1）掌握如何利用汇编语言设计分支程序、循环程序和子程序；

2）掌握堆栈在子程序设计中的重要作用；

3）熟悉程序优化的一般方法。

本章教学支持的课程目标为目标1和目标2。

**2.教学重点**

1）转移指令

理解各类转移指令的功能、条件。

2）子程序

子程序的定义、调用、返回，参数传递方法。

3）堆栈

掌握堆栈在子程序返回、参数传递、现场保护、局部变量、程序设计等环节中的重要作用。

**3.教学难点**

1）转移指令的选择

掌握如何根据上下文、寻址方式的要求选择合适的转移指令达到正确转移的目的。

2）子程序的设计

准确把握调用与返回指令的原理，与堆栈的运用有机结合起来。

**4.教学环节设计**

围绕教学重点和教学难点，综合应用讨论、作业、实验、共享资源学习等教学形式。

1）讨论

本章讨论围绕转移指令的选择等开展。

2）作业

围绕编写分支、循环、子程序的基本框架和熟悉转移指令、堆栈操作等内容布置，并与实验环节相结合。

3）实验

配合《汇编语言程序设计实践》课程完成相关程序的设计与调试，观察转移指令、子程序、堆栈等的操作特点。

4）共享资源学习

完成本课程精品资源共享课平台相关内容的学习。

**5.思政元素**

循环程序中最主要的特点是不断重复可能并不复杂的事，直到达到预期的目的后才结束。这类程序能体现量变与质变的关系，以及坚持不懈、不达目的誓不罢休的努力奋斗的精神。

**第五章 宏功能、模块化程序设计**

本章的主要知识点包括宏功能程序设计，模块化程序设计。通过本章教学让学生认识到提高编程语言编写复杂程序的效率与程序可读性的必要性和基本思路，了解多个程序模块组装成一个程序的基本原理。

**1.教学目标**

1）掌握宏指令的作用，宏指令的定义、调用、展开的方法；

2）掌握模块化程序设计的基本方法，了解模块组装的基本原理。

本章教学支持的课程目标为目标1和目标2。

**2.教学重点**

1）宏指令的定义与调用

2）模块化程序设计中的通讯方式

**3.教学难点**

1）段的组合方式

2）与C语言混合编程时的方法

通过尝试与C语言的模块进行组装，了解不同语言的约定对程序组装成功的影响。

**4.教学环节设计**

围绕教学重点和教学难点，除课堂讲授外，还需综合应用作业、实验、共享资源学习等教学形式。

1）作业

围绕如何利用宏功能改善程序的可读性等方面布置，并与实验环节相结合。

2）实验

配合《汇编语言程序设计实践》课程完成相关程序的设计与调试，观察宏的展开与段的组合情况等。

3）共享资源学习

完成本课程精品资源共享课平台相关内容的学习，了解混合编程的一些实践经验。

**5.思政元素**

模块化程序设计要求为每个模块定义好与其他模块对接的信息，来不得半点马虎，这能培养学生之间团结、合作的意识。

**第六章 输入输出和其他编程问题**

本章的主要知识点包括输入输出指令，中断的基本概念及中断处理程序的设计，不同运行环境下程序的设计问题等。通过本章的学习，了解CPU与外部设备打交道的方法，中断的基本原理，掌握中断处理程序的设计方法，了解16/32/64位等不同环境下的程序设计特点。

**1.教学目标**

1）掌握输入输出指令；

2）掌握中断的基本概念及中断处理程序的设计方法；

3）了解16/32/64位程序的编程特点。

本章教学支持的课程目标为目标1和目标2。

**2.教学重点**

1）中断的基本概念及中断处理程序的设计方法

中断的响应过程，中断矢量表的结构与作用，中断处理程序编写与安装方法。

2）16/32/64位程序的主要特点

**3.教学难点**

1）中断的响应过程

通过对中断硬件、中断矢量表、中断处理程序、软中断指令、中断返回指令的理解来把握中断的响应过程。

2）中断程序的部署

如何合理安排中断处理程序，才能保证中断处理程序不受其他程序的影响。

**4.教学环节设计**

围绕教学重点和教学难点，除课堂讲授外，还需综合应用作业、实验、共享资源学习等教学形式。

1）作业

围绕中断响应过程涉及到的内容布置，并与实验环节相结合。

2）实验

配合《汇编语言程序设计实践》课程完成相关程序的设计与调试，观察中断矢量表、中断的响应过程；观察16/32/64位程序的特点等。

3）共享资源学习

通过本课程精品资源共享课平台相关内容的学习以及查找网络资料，了解16/32/64位编程的一些实践经验。

**5.思政元素**

中断处理分级与计算机安全中信息安全事件分级的类比讲解，让学生了解信息安全事件及响应的一般知识，增强信息安全与社会管理方面的意识。

九、教与学

**1.**教学方法

主要的教学环节包括课堂授课、讨论、作业、共享资源学习和配套实验等环节。本课程的教学设计特色主要体现在如下三个方面：

1）在本课程的教学过程中将采用基于问题的教学方法，精心设计若干探究性问题，引导学生深入思考，加深所学知识的理解和应用，提高学习效果。

2）本课程的教学过程中，要牢固树立软硬件协同工作的观点。首先，要把“汇编语言程序设计”的学习与“数字逻辑”、“C语言”等先学课程紧密联系起来，用已学的知识理解汇编语言，用汇编语言反思已学的知识，同时，也要把汇编语言的学习与后续课程“计算机组成原理”、“接口技术”、“操作系统”、“编译原理”等联系起来，打开学生学习的视野；其次，要注重把各章节的知识有机结合，共同构成一个完整的计算机系统，如在CPU内标志寄存器的学习过程中，应结合指令之间如何关联以及条件转移指令的使用等知识；在学习中断的过程中，要结合CPU的结构、输入输出、子程序、跟踪与反跟踪、BIOS和操作系统调用等相关内容；第三，实例演示，化抽象为具体，如利用调试工具或仿真软件让学生看得到程序或者指令的执行过程，看得到寄存器、内存等等的变化，看得见高级语言程序编译后的结果，使得抽象的计算机工作原理能有一个具体的体现，便于理解；第四，结合日常生活中的例子或游戏，变枯燥乏味为生动有趣，例如，寻址方式概念，可以与日常生活中寻找一个人的方法相比较，使得学生能较直观的认识到寻址方式的概念。第五，在教学过程中，要注重建立汇编语言在解决计算机应用问题中的特别之处，如编写超短程序的技巧，编写超快程序的方法，改变计算机工作状态的方法，灵活实现程序功能的方法，跟踪与反跟踪的方法，等等，力求通过逻辑思维和发散思维的训练，培养学生的创新思维能力，使学生深入理解计算机的工作原理，提升软件设计水平。

3）强调动手实践。该课程的教学与独立设置的课程实验相配合，实验内容与理论课程教学进度同步，通过实验加深对所学理论知识的理解，通过实验也可以检验理论课程的学习效果。

2.学习方法

本课程是一门语言课，也是一门原理课，因此，它具有理论性、技术性和实践性。学习过程中，首先要注重语言的基本规律与规则，同时要理解支撑语言所需的计算机基本结构、基本原理；其次，要掌握用汇编语言编写基本程序的方法，但不能以完成程序的功能为最终目的，而要以深入理解汇编语言的特点与优势、计算机软硬件协同工作的原理、本课程的相关原理与方法如何进一步支撑软件系统高效运行以及高级语言的处理等为最终目标进行思考与升华。

学生要积极参与讨论、深刻理解原理和技术本质，也需要亲自动手完成作业和课程配套开设的实验。通过实验，加强对课程理论知识的理解，同时，训练学生发现问题、分析问题和解决问题的能力；学会通过共享资源平台和网络资源，从不同角度理解知识，获取其他学习者分享的学习经验。

十、学时分配

|  |  |  |
| --- | --- | --- |
| **序号** | **主要内容** | **学时分配** |
| 1 | 第一章 计算机基本工作原理 | 5 |
| 2 | 第二章 寻址方式 | 2 |
| 3 | 第三章 宏汇编语言 | 5 |
| 4 | 第四章 程序设计的基本方法 | 5 |
| 5 | 第五章 宏功能、模块化程序设计 | 2 |
| 6 | 第六章 输入输出和其他编程问题 | 5 |
| 总计 | | 24 |

十一、课程考核与成绩评定

1.课程成绩构成

课程最终成绩由平时作业成绩、课程期末考试成绩综合而成，各部分成绩的比例如下：

**1）平时作业成绩：30%。**包括课后作业和课堂作业。将引导学生有针对性地复习和巩固讲授的内容，思考和解决一些不太复杂或有灵活性要求的问题，以及有准备地参与到讨论中并形成记录（作业）。主要考察作业完成率和完成质量。

**2）期末考试成绩：70%。**主要考核本课程基础知识和基本能力的掌握程度，是对学生学习情况的全面检验。考试强调对本课程的基本概念、基本方法和技术的掌握，也即利用汇编语言的方法描述和求解计算机工程问题的基本能力的掌握情况，同时也会要求学生能利用汇编语言不同于其他语言的特点，编写出具有汇编语言优势的软件功能模块；并通过多种综合型试题考核学生运用所学知识解决复杂工程问题的能力。考试采用书面闭卷考试形式。主要题型为填空、改错、简答、分析、设计等题型。

课程考核成绩评定如表1所示。

表1 汇编语言程序设计课程考核与成绩评定

|  |  |  |  |
| --- | --- | --- | --- |
| **课程目标** | **考核与评价方式及成绩比例** | | **成绩占比(约）** |
| **平时作业** | **期末考试** |
| 1 | 20 | 49 | 69% |
| 2 | 10 | 21 | 31% |

2.考核与评价标准

1）平时作业成绩考核与评价标准

表2课程目标1平时作业考核与成绩评定

|  |  |  |  |
| --- | --- | --- | --- |
| **评价标准** | | | |
| **优秀** | **良好** | **及格** | **不及格** |
| 按时提交作业，汇编语言的基本知识和基本原理的理解与应用准确，解决方案描述清晰，汇编语言源程序功能简洁正确。 | 按时提交作业，汇编语言的基本知识和基本原理的理解与应用较准确，解决方案描述较清晰，汇编语言源程序功能比较简洁正确。 | 按时提交作业，汇编语言的基本知识和基本原理的理解与应用基本准确，解决方案描述有些模糊，汇编语言源程序功能存在少量错误。 | 未按时交作业，概念与原理理解欠准确，结果错误较多。 |

表3课程目标2平时作业考核与成绩评定

|  |  |  |  |
| --- | --- | --- | --- |
| **评价标准** | | | |
| **优秀** | **良好** | **及格** | **不及格** |
| 按时提交作业，对基本原理理解深入，能利用汇编语言的特点灵活地给出解决方案，结果正确，分析过程充分，论述清晰。 | 按时提交作业，对基本原理理解较深入，能利用汇编语言的特点比较灵活地给出解决方案，结果正确，分析过程较充分，论述较清晰。 | 按时提交作业，能利用汇编语言的特点给出解决方案，结果存在少量错误，能给出一定的分析论述。 | 未按时交作业，对汇编语言的特点把握不清，结果错误较多。 |

2）课程考核与成绩评定

根据期末考试的试卷评分标准进行评定。

汇编语言程序设计课程组

2015年6月制定

2019年7月修订

2020年1月修订

2020年9月修订

2021年5月修订

2021年7月修订

2022年2月修订

