**《算法与数据结构》教学大纲**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 课程英文名 | Algorithm Analysis and Data Structures | | | | 课程代码 | 03M0132 | |
| 学分 | 3 | 总学时 | 48 | 理论学时 | 36 | 实验/实践学时 | 12 |
| 课程类别 | 学科基础课 | 课程性质 | 任选 | 先修课程 | 高等数学、线性代数、C语言程序设计、面向对象程序设计 | | |
| 适用专业 | 电子信息工程 | | | 开课学院 | 信息工程学院 | | |
| 执笔人 | 项伟平 | 审定人 | 肖丙刚 | 制定时间 | 2020 年11 月 | | |

**一、课程地位与课程目标**

（一）课程地位

本课程是电子信息工程专业的一门重要的学科基础选修课程。本课程应用高等数学的微积分知识、线性代数的向量和矩阵的概念，在掌握C语言和面向对象程序设计基础上，研究数据表示及数据处理以及算法运行时间复杂度的估算，其任务主要是要求学生掌握常用数据结构的基本概念及其不同的实现方法，通过系统学习能够在不同存储结构上实现不同的运算，并对算法设计进行有效的分析，进一步提升学生程序设计的能力。有助于学生获取描述电子信息工程领域复杂工程问题的工程基础知识；求解电子信息工程问题的模型，并对解决方法进行评价；完成设计方案并分析阐明设计的合理性。由于是双语教学，肯定有助于提高学生的专业英语能力，从而能够就本专业的当前热点问题发表自己的想法，进行国际交流。同时，本课程为后续课程、毕业设计以及从事软件开发方面的工作打下坚实的基础。

（二）课程目标

1. 掌握算法分析的数学基础，具备对不同算法进行运行时间复杂度估算并比较其效率的能力。

2. 掌握线性数据结构原理与实现，具备线性数据结构在实际算法中的应用能力。

3. 掌握非线性数据结构原理与实现，具备非线性数据结构在实际算法中的应用能力。

4. 理解各种排序算法原理与实现，具备比较不同排序算法运行效率的能力。

5. 理解和掌握专业英语词汇，初步具备阅读原版教科书、进行国际交流的能力。

6. 培养工程师的基本素养，寻求和坚持真理，在将来的工作中，杜绝有所欺骗的行为，诚实、公平、忠实地为雇主和客户服务。

**二、课程目标达成的途径与方法**

主要以课堂教学为主，结合期末测试、课堂测试、课后作业和实验等途径和方法来达成。

**三、课程目标与相关毕业要求的对应关系**

|  |  |  |  |
| --- | --- | --- | --- |
| 课程目标 | 课程目标对毕业要求的支撑程度（H、M、L） | | |
| 毕业要求1.1 | 毕业要求2.2 | 毕业要求10.3 |
| 课程目标1 | L(0.05) |  |  |
| 课程目标2 |  | H(0.25) |  |
| 课程目标3 |  | H(0.25) |  |
| 课程目标4 |  | H(0.25) |  |
| 课程目标5 |  |  | M(0.15) |
| 课程目标6 | L(0.05) |  |  |

**四、课程主要内容与基本要求**

1、Introduction

Understand: the need for data structures

Understand: the relationship between abstract data types and data structures

Master: C++ review

Command: mathematical preliminaries such as sets and recursion

这部分是本课程的基础部分，复习C++的基本概念，引入集合、递归等数学预备知识，为后续章节的学习做准备。

2、Algorithm Analysis

Command: the definition of algorithm analysis

Understand: the growth rate for an algorithm

Command: best-case/worst-case/average-case analysis of an algorithm

Master: big-Oh, big-Omega, and big-Theta notations

Master: how to calculate the running time of a program

这部分是引入算法和算法分析的概念，并建立经典的数学模型分析算法的时间复杂度。

3、Lists, Stacks, and Queues

Master: the ADT for a list

Command: different implementations for a list

Master: the ADT for a stack

Command: different implementations for a stack

Master: the ADT for a queue

Command: different implementations for a queue

这部分是重点分析了三种基本的数据结构（表、栈和队列），详细分析了其抽象数据类型和各种不同的实现方法，通过对各种算法的分析，能使学生熟悉程序设计的技巧。

4、Binary Trees

Master: definitions and terminology of a binary tree

Command: binary tree traversals

Command: pointer-based binary tree node implementations

Master: binary search trees

Understand: AVL trees

这部分是重点分析了二叉树结构，详细分析了其抽象数据类型和各种不同的实现方法，重点介绍了具有很强实用价值的AVL数结构。通过对各种算法的分析，能使学生熟悉程序设计的技巧。

5、Sorting

Master: sorting terminology and notation

Command: insertion sort, bubble sort, and selection sort

Command: shellsort, quicksort, mergesort, heapsort

Understand: comparison of sorting algorithms

Master: calculate the running time of a program

这部分是详细介绍了各种经典的排序算法。在第二章的基础上，对不同的排序算法进行分析，能够针对不同的应用场合选择最优的算法。

**五、课程学时安排**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 章节号 | 教学内容 | 学时数 | 学生任务 | 对应课程目标 |
| 1 | Introduction | 6 | Exercises 1.3, 1.6 | 课程目标1  课程目标5  课程目标6 |
| 2 | Algorithm Analysis | 8（Include：2 hours’ lab. assignment） | Exercises 2.2, 2.11 | 课程目标1  课程目标5  课程目标6 |
| 3 | Lists, Stacks, and Queues | 14（Include：6 hours’ lab. assignment） | Exercises 3.2, 3.4, 3.28 | 课程目标2  课程目标5  课程目标6 |
| 4 | Binary Trees | 16（Include：4 hours’ lab. assignment） | Exercises 4.1, 4.3, 4.8, 4.9 | 课程目标3  课程目标5  课程目标6 |
| 5 | Sorting | 4 |  | 课程目标4  课程目标5  课程目标6 |

**六、实践环节及基本要求**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 序号 | 实验项目名称 | 学时 | 基本要求 | 学生任务 | 实验性质 | 实验类别 |
| 1 | Lab. Assignment 1 Application of Structures—Introduction to VC++ IDE | 2 | 学会使用VC++ IDE开发应用程序 | 利用结构存储员工的工资、ID等，并进行排序 | 验证性 | 必做 |
| 2 | Lab. Assignment 3 Application of a Stack—Reversing a line | 2 | 学会创建并使用栈结构 | 建立栈结构，并应用栈结构实现一行字符的反向输出 | 设计性 | 必做 |
| 3 | Lab. Assignment 2 Application of a Linked List—Addition of Polynomials | 4 | 学会创建并使用链表 | 建立链表，存储两个多项式，然后对两个多项式进行加法运算，最后显示和多项式 | 综合性 | 必做 |
| 4 | Lab. Assignment 8 Application of a binary tree—a binary tree for Stemma | 4 | 学会创建并使用二叉树 | 建立二叉树结构，存储家谱，并按层次结构打印家谱 | 综合性 | 必做 |

**七、考核方式及成绩评定**

|  |  |  |  |
| --- | --- | --- | --- |
| 考核内容 | 考核方式 | 评定标准（依据） | 占总成绩比例 |
| 过程考核 | 课堂点名、作业、实验 | 到课率、作业批改成绩、实验操作和实验报告成绩 | 30% |
| 期末考核 | 闭卷 | 卷面成绩 | 70% |
| 考核类别 | 考查 | | |
| 成绩登记方式 | 百分制 | | |

**八、课程目标达成度评价方法**

|  |  |  |
| --- | --- | --- |
| 课程目标 | 教学环节 | 成绩评定 |
| 课程目标1 | 讲授 | 作业A10 |
|  |  | 期末考试B10 |
|  | 实验一 | 实验成绩C10 |
| 目标达成度1=（0.1☓A1+0.7☓B1+0.2☓C1）/（0.1☓A10+0.7☓B10+0.2☓C10） | |
| 课程目标2 | 讲授 | 作业A20 |
| 期末考试B20 |
| 实验二、实验三 | 实验成绩C20 |
| 目标达成度2=（0.1☓A2+0.7☓B2+0.2☓C2）/（0.1☓A20+0.7☓B20+0.2☓C20） | |
| 课程目标3 | 讲授 | 作业A30 |
| 期末考试B30 |
| 实验八 | 实验成绩C30 |
| 目标达成度3=（0.1☓A3+0.7☓B3+0.2☓C3）/（0.1☓A30+0.7☓B30+0.2☓C30） | |
| 课程目标4 | 讲授 | 期末考试B40 |
| 目标达成度4=B4/B40 | |
| 课程目标5 | 讲授 | 期末考试B50 |
| 目标达成度5=B5/B50 | |
| 课程目标6 | 讲授 | 期末考试B60 |
| 目标达成度6=B6/B60 | |

**九、推荐教材与主要参考书**

（一）推荐教材：

1、《数据结构与算法分析：C++语言描述（第四版）》. Mark Allen Weiss. 电子工业出版社，2017年8月。

（二）主要参考书：

1、Clifford A. Shaffer.《A PRACTICAL INTRODUCTION TO DATA STRUCTURES AND ALGORITHM ANALYSIS (C++ EDITION)》. Second Edition，电子工业出版社，2009年9月；

2、实验参考书：《软件技术基础实验讲义》，自编。