**《数据结构》教学大纲**

**一、课程基本信息**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **课程编号** | 4MSCE041036 | | **课程中文名称** | | 数据结构 | | | | |
| **培养方案版本** | 2018版 | | **课程英文名称** | | Data Structure | | | | |
| **开课学期** | 2 | | **学分** | | 4 | | | **总学时** | 64 |
| **理论学时** | 54 | | **实验学时** | |  | | | **上机学时** | 10 |
| **适用专业** | ☑计算机科学与技术 □电子信息工程 □网络工程 □软件工程 | | | | | | | | |
| **课程类别** | □人文社科 □公共基础 🗹学科基础 □专业必修课 □专业选修课 □素质教育 □实践教学环节 | | | | | | | | |
| **修读方式** | □公共必修课 □公共选修课 ☑专业必修课 □专业选修课 | | | | | | | | |
| **先修课程** | 《程序设计基础》 | | | | | | | | |
| **课程负责人** | 党源源 | **职 称** | | 副教授 | | **课程组成员** | 王红梅、王新颖、王小虎、刘冰、索冬梅 | | |
| **执笔人** | 王红梅 | **审核人** | | 党源源 | | **修订时间** | 2016年5月 | | |

**二、课程简介**

|  |  |
| --- | --- |
| **类别** | **描述** |
| 课程定位 | 数据结构是计算机科学与技术、软件工程等专业的专业基础课程，在整个课程体系中处于承上启下的核心地位，作为程序设计语言后续课程的数据结构课程是对程序设计语言的重要补充和发展，同时为进一步学习操作系统、编译原理、数据库等专业课奠定坚实的基础。 |
| 主要内容 | 线性表、栈和队列、字符串和多维数组、树和二叉树、图的逻辑结构、存储结构、基本操作及经典算法，查找和排序的数据处理技术。 |
| 学习成果 | 理解各种数据结构的实现方法，针对具体工程问题分析问题逻辑结构，选择、设计存储结构，并在此基础上应用程序设计语言实现基本操作。理解各种经典算法的实现过程，可复现数据结构中的经典算法，进行初步算法时间、空间分析。在解决问题过程中，形成初步计算思维能力、算法设计能力、运用程序设计语言解决实际问题的能力。 |

**三、课程目标**

本课程的主要教学环节有理论教学、实验教学，课程教学的主要目标：理解表、树、图基本数据结构逻辑结构、存储结构、基本操作的构建、实现过程，针对具体工程问题构建数据模型，比较、设计存储结构，实现基本操作算法；理解数据结构经典算法及常用的数据处理技术经典算法，针对具有时间性能约束、空间性能约束的复杂工程问题选出合适存储结构及算法，形成基本解决方案，进行初步算法分析；理论知识与实践教学相结合，在实践过程中初步形成计算思维能力、运用程序设计语言解决实际问题的能力。具体教学目标如下：

1．理解基本数据结构逻辑结构、存储结构、基本操作的构建、实现过程，针对受时间、空间约束的计算机工程问题，应用数据结构的基本原则和方法，构建合适的数据模型，通过比较、选择、优化设计合理的存储结构，并进行评价。

2．理解数据结构经典算法及常用的数据处理技术经典算法，针对具有时间性能约束、空间性能约束的计算机工程问题，构建合适的数据模型，选择不同的经典算法，进行算法复现及重组，并进行时间、空间的定性渐近分析，依据对比结果形成初步的解决方案。

3．针对具体工程问题，应用“问题🡺想法🡺算法🡺程序”的问题求解过程及一般方法，进行程序设计实现问题求解，并在求解过程中体现创新意识，培养基于算法设计、算法实现的解决实际问题的能力。

1. **课程支撑的毕业要求指标点及权重**

|  |  |  |  |
| --- | --- | --- | --- |
| **毕业要求** | **毕业要求指标点** | **课程目标** | 权重 |
| 毕业要求1 | 1.4 能够运用工程知识、专业知识和数学模型方法，对计算机领域复杂工程问题的解决方案进行比较与综合 | 课程目标2 | 0.3 |
| 毕业要求2 | 2.3 能够认识到解决计算机领域工程问题有多种方案可以选择，会通过文献研究寻求可替代的解决方案 | 课程目标1 | 0.3 |
| 毕业要求2 | 2.4 能够运用计算机工程科学的基本原理，借助文献研究，分析方案执行过程的影响因素，并获得有效结论 | 课程目标3 | 0.4 |

**五、教学方法**

1. 问题驱动的课堂讲授法。课程的总体教学目标是培养学生的计算思维能力和应用计算机解决问题的算法与程序设计能力，因此，将教学主线确定为计算机进行问题求解的一般过程“问题🡪想法🡪算法🡪程序”，在这个一般过程中通过设疑、解疑，采用互动的方式进行知识传递。

2. 基于图示的动画演示法。在深刻揭示知识内涵的基础上，并将教学意图融入多媒体课件，将教学内容中抽象的概念和不可见的过程，通过动画形象地动态演示出来。

3. 引导思维的启发式教学。注意引导思维，通过讲思路讲过程讲方法，展现问题的求解过程，通过设疑、问疑、解疑一步步启发学生的思维，并将启发式教学方法贯彻到每一个教学专题。

4. 项目驱动的讨论式教学。根据课程内容布置具有一定难度的算法问题（结合ACM程序设计竞赛），以小组方式进行讨论，以抽签方式进行展示，提高学生的主动学习意识。

5. 全程跟踪的混合式教学。采用互动式教学工具，课前向学生发送预习任务，课中设计相关测验完成即时考核，提高课堂的有效学习时间，课后完成问卷调查和学习检查。

**六、教学环节与教学安排**

**1.理论环节教学安排**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **序号** | **知识单元** | **教学内容** | **学时分配** | **基本要求（能力）** | **课程目标** |
| 1 | 绪论 | 课程概述；问题求解与程序设计；数据结构的基本概念；算法的基本概念；算法分析。重点是数据结构的基本概念；数据的逻辑结构、存储结构以及二者之间的关系；算法及算法的特性；大*Ｏ*记号。难点是抽象数据类型；算法的时间复杂度分析。 | 5 | 理解本课程的重要性，界定课程目标及学习方法；理解数据结构和算法在程序设计中的作用；理解数据结构的基本概念；理解算法的基本概念，具有算法分析能力。 | 目标1 |
| 2 | 线性表 | 线性表的逻辑结构；线性表顺序存储结构及实现；线性表链接存储结构及实现；顺序表和链表的比较。重点是顺序存储结构和链接存储结构的基本思想；顺序表和单链表的基本算法；顺序表和单链表基本操作的时间性能；顺序表和链表之间的比较。难点是线性表的抽象数据类型定义；基于单链表算法设计；双链表算法设计。 | 6 | 针对线性表问题可设计其逻辑结构、设计顺序表实现其算法、设计单链表实现其算法，理解其他线性表链接存储结构及算法实现，比较顺序表和链表各自特性。 | 目标1  目标2  目标3 |
| 3 | 栈和队列 | 栈的逻辑结构定义；栈的存储结构及实现；队列的逻辑结构定义；队列的存储结构及实现。重点是栈和队列的操作特性；栈和队列基本操作的实现。难点是循环队列的存储方法；循环队列队空和队满的判定条件 | 4 | 对具体问题依据栈和队列的操作特性选择正确的数据结构；设计顺序栈存储结构及实现；设计链栈存储结构及实现；设计循环队列存储结构及实现；设计链队列存储结构及实现 | 目标1  目标2  目标3 |
| 4 | 字符串和多维数组 | 字符串的逻辑结构、存储结构及其串上的基本运算；数组的逻辑结构特征、存储结构及寻址；特殊矩阵和稀疏矩阵的压缩存储方法。重点是串的模式匹配算法；数组的寻址方法；特殊矩阵的压缩存储和寻址方法。难点是KMP算法；稀疏矩阵的压缩存储 | 4 | 理解字符串的逻辑结构；理解字符串的存储结构；针对具体问题可复现模式匹配BF算法；理解模式匹配KMP算法；可设计数组的逻辑结构；设计数组的存储结构，实现寻址；理解特殊矩阵的压缩存储；理解稀疏矩阵的压缩存储 | 目标1  目标2  目标3 |
| 5 | 树和  二叉树 | 树的逻辑结构；树的存储结构；二叉树的逻辑结构；二叉树的存储结构及实现；树、森林和二叉树之间的转换；哈夫曼树及哈夫曼编码。重点是二叉树的性质；二叉树和树的存储表示；二叉树的遍历及算法实现；树与二叉树的转换关系；哈夫曼树。难点是二叉树的层序遍历算法；二叉树的建立算法；哈夫曼算法 | 9 | 针对树问题可设计树的逻辑结构和存储结构；可以应用树和二叉树的遍历解决问题；理解二叉树的定义，应用二叉树基本性质解决问题；设计并实现二叉链表及其遍历递归算法；理解二叉树的其他存储结构和非递归算法；可应用二叉链表层序遍历及建立算法解决实际问题；可对树、森林和二叉树之间的转换进行推导；可应用哈夫曼树及哈夫曼编码解决实际问题。 | 目标1  目标2  目标3 |
| 6 | 图 | 图的逻辑结构；图的存储结构及实现；最小生成树；最短路径；有向无环图。重点是图的基本术语；图的存储表示；图的遍历；图的经典应用。难点是图的遍历算法；Prim算法；Kruskal算法；Dijkstra算法；Floyd算法；拓扑排序算法；关键路径算法 | 12 | 记住图的定义及基本术语；设计图的邻接矩阵存储、实现算法；设计图的邻接表存储、实现算法；可复现Prim算法、Dijkstra算法、拓扑排序算法；理解Kruskal算法、Floyd算法、AOV网的定义及性质、AOE网的定义及性质、关键路径算法，可据具体问题进行算法选择。 | 目标1  目标2  目标3 |
| 7 | 查找技术 | 查找的基本概念及算法性能；线性表的查找技术；树表的查找技术；散列表的查找技术；各种查找方法的比较。重点是折半查找算法及性能分析；二叉排序树的构造及查找；平衡二叉树的调整；散列表的构造和查找；B树的定义。难点是二叉排序树的删除操作；平衡二叉树的调整；B树的插入和删除操作 | 7 | 理解查找的基本概念及算法性能；可实现并应用顺序查找算法、掌握折半查找、二叉排序树的插入；理解二叉排序树的删除；可进行二叉排序树的查找及性能分析、平衡二叉树的调整、B树的定义、B树的查找；理解B树的插入和删除；理解常见的散列函数、散列查找的性能分析；设计并实现散列的构造和查找；比较各种查找方法特性。 | 目标2  目标3 |
| 8 | 排序技术 | 排序的基本概念及算法性能；插入排序；交换排序；选择排序；归并排序；各种排序算法的比较。重点是各种排序算法的基本思想；各种排序算法的执行过程；各种排序算法及时间复杂度分析；各种排序算法之间的比较。难点是快速排序、堆排序、归并排序等算法及时间复杂度分析 | 7 | 理解排序的基本概念；分析排序算法的性能；设计并实现直接插入排序、气泡排序、快速排序的一次划分算法、简单选择排序；实现快速排序、堆排序、二路归并排序的非递归实现、二路归并排序的递归实现；理解希尔排序、筛选法调整堆算法；比较各种排序算法特性。 | 目标2  目标3 |

**2.实践环节教学安排**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **序号** | **实验内容** | **学时分配** | **基本要求（能力）** | **课程目标** |
| 1 | 线性表及其应用：实际问题中线性表的插入和删除（顺序表、链表） | 2 | 针对线性表实际问题抽象数据模型、设计存储结构、进行算法设计与程序实现，进行实验总结，撰写实验报告。 | 目标3 |
| 2 | 栈和队列及其应用：运用栈来模拟程序的递归过程，实现*n*阶汉诺塔问题 | 2 | 针对栈、队列实际问题抽象数据模型、设计存储结构、进行算法设计与程序实现，进行实验总结，并撰写实验报告。 | 目标3 |
| 3 | 树及其应用：建立一棵二叉树、遍历二叉树各结点（按层次、前序、中序、后序） | 2 | 针对树实际问题抽象数据模型、设计存储结构、进行算法设计与程序实现，进行实验总结，并撰写实验报告。 | 目标3 |
| 4 | 图及其应用：建立图的邻接矩阵和邻接表，针对实际问题求最小生成树、最短路径 | 2 | 针对图实际问题抽象数据模型、设计存储结构、进行算法设计与程序实现，进行实验总结，并撰写实验报告。 | 目标3 |
| 5 | 查找、排序算法的比较：给出具有不同特性的数据，采用各种查找方法对给定关键码进行查找，比较查找性能；采用各种排序方法对其进行排序，比较排序性能 | 2 | 针对查找、排序实际问题抽象数据模型、设计存储结构、进行算法设计与程序实现，进行实验总结，并撰写实验报告。 | 目标3 |

**七、教材、参考书目和教学资源**

**教材：**

[1] 王红梅等编著.数据结构——从概念到C实现（第2版）.北京：清华大学出版社，2017.

[2] 数据结构（C++版）学习辅导和实验指导（第2版）.王红梅等编著.清华大学出版社，2014.

**参考书目：**

[1] Thomas H. Corman 等.殷建平等译.算法导论（第3版）. 机械工业出版社. 2013.

[2] Mark A. Weiss. 数据结构与算法分析：C++语言描述（第2版）. 机械工业出版社. 2010.

[3] Bentley .黄倩译.编程珠玑. 人民邮电出版社. 2011.

[4] 数据结构(C语言版). 严蔚敏等编著. 北京：清华大学出版社. 2010.

[5] 数据结构(C++语言版)第3版. 邓俊辉编著. 北京：清华大学出版社. 2013.

**教学资源：**

**MOOC 课程**

1. 耿国华. 西北工业大学：<http://www.icourse163.org/course/NWU-298002>
2. 陈越. 浙江大学：https://www.icourse163.org/course/zju0901-93001/

**八、考核方案**

**1.课程目标达成途径**

|  |  |  |  |
| --- | --- | --- | --- |
| **课程目标** | **达成途径** | **考核内容** | **考核方式** |
| 课程目标1 | 驱动式课堂教学：以“问题”为驱动，引导学生从问题出发，进行逻辑模型构建；进行问题回归，再次回到问题，在其中找出依据，进行存储结构选择、重组。  课外作业：通过完成布置的习题，巩固数据结构逻辑结构、存储结构、基本操作的原理知识，形成基本应用能力。  随堂小测：针对数据结构逻辑结构、存储结构、基本操作相关知识选取选择题型，以随堂发放考核形式，检查出勤情况以及课堂掌握情况，促进听课效果。  期中考试：通过闭卷考试，阶段性检验具体问题下逻辑模型构建、存储结构选择、基本操作实现应用能力。  期末考试：通过闭卷考试，全面检验具体问题下逻辑模型构建、存储结构选择、基本操作实现应用能力。 | 1.针对给出问题，进行表、树、图等数据结构逻辑模型的构建；  2.针对给出问题，在问题条件约束下选择与优化存储结构。 | 随堂小测，  课外作业，  期中考试，  期末考试。 |
| 课程目标2 | 驱动式课堂教学：以“问题”为驱动，引导学生从问题出发，形成“想法”，结合存储结构完成由“想法”到“算法”的实践过程，并进行算法复杂度分析。  课外作业：通过完成布置的习题，巩固数据结构经典算法及常用的数据处理技术经典算法，形成基本复现、应用能力。  随堂小测：针对数据结构经典算法及常用的数据处理技术经典算法知识选取选择题型，以随堂发放考核形式，检查出勤情况以及课堂掌握情况，促进听课效果。  期中考试：通过闭卷考试，阶段性检验时间性能约束、空间性能约束的复杂工程问题中，算法设计能力；检验进行时间、空间定性渐近分析能力。  期末考试：通过闭卷考试，全面检验时间性能约束、空间性能约束的复杂工程问题中，选择经典算法的设计思想，进行算法复现及重组的能力；检验进行时间、空间定性渐近分析能力。 | 1. 针对给出问题，选择并应用数据结构经典算法及常用的数据处理技术经典算法进行问题求解； 2. 针对给出问题，选择满足时间、空间性能约束的典算法，并对求解过程进行时间、空间复杂度分析； | 随堂小测，  课外作业，  期中考试，  期末考试。 |
| 课程目标3 | 实验教学：针对基本数据结构及数据处理技术的问题，采用“问题🡺想法🡺算法🡺程序”问题求解过程及一般方法，开展主题实验，形成初步计算思维能力、运用程序设计语言解决实际问题的能力。  期中考试：通过闭卷考试，阶段性检验面对工程问题，采用“问题🡺想法🡺算法🡺程序”问题求解方法进行问题求解能力；考察算法设计及运用程序设计语言解决实际问题的能力。  期末考试：通过闭卷考试，全面检验面对复杂工程问题，采用“问题🡺想法🡺算法🡺程序”问题求解方法进行问题求解能力；考察算法设计及运用程序设计语言解决实际问题的能力。 | 针对工程实践具体问题，应用“问题🡺想法🡺算法🡺程序”的问题求解过程及一般方法，进行逻辑模型设计、数据结构设计实现、算法设计与程序设计，完成问题求解。 | 实验教学，  期中考试，  期末考试。 |

**2.考核环节与成绩比例**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **课程目标** | **作业（10%）** | **随堂小测（10%）** | **期中考试（10%）** | **实验（10%）** | **期末考试（60%）** | **权重和** |
| **课程目标1** | 0.1 | 0.1 | 0.2 |  | 0.6 | 1 |
| **课程目标2** | 0.1 | 0.1 | 0.2 |  | 0.6 | 1 |
| **课程目标3** |  |  | 0.2 | 0.6 | 0.2 | 1 |

**九、考核标准依据**

**1.作业考核与评价标准**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 课程目标 | 评价标准 | | | | | |
| 100~90分 | 89~80分 | 79~70分 | 69~60分 | 59~20分 | 19~0分 |
| 目标1 | 针对给出问题可分析得出正确逻辑模型；针对逻辑模型可对所学存储结构进行比较、组合、优化，设计出合适存储结构 | 针对给出问题可分析得出正确逻辑模型；针对逻辑模型可对所学存储结构进行比较，结合比较结果选择合适的存储结构 | 针对给出问题可分析得出正确逻辑模型；可针对逻辑模型选择合适的存储结构。 | 针对给出问题可分析得出正确逻辑模型；可针对逻辑模型给出一种基本存储结构。 | 针对给出问题可分析得出基本逻辑模型。 | 不能进行基本逻辑模型分析、构建。 |
| 目标2 | 针对给出问题可选择合适的经典算法或设计时间性能、空间性能较优的算法进行“问题”解决；可对算法进行时间、空间性能分析，并据分析结果进行算法改进 | 针对给出问题可选择合适的经典算法或设计满足时间性能、空间性能要求的算法进行“问题”解决；可对算法进行时间、空间性能分析。 | 针对给出问题可选择合适的经典算法或设计基本算法解决“问题”；可对算法进行基本的时间、空间性能分析。 | 针对给出问题可选择经典算法或设计基本算法解决“问题”。 | 针对给出问题可给出解决“问题”基本思路。 | 针对给出问题不能给出解决“问题”基本思路。 |

注：每次作业均以百分计，作业成绩70分达到课程目标。

**2.随堂小测考核与评价标准**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 课程目标 | 评价标准 | | | | | |
| 100~90分 | 89~80分 | 79~70分 | 69~60分 | 59~20分 | 19~0分 |
| 目标1 | 针对考核问题，在时间限定内，可分析得出正确逻辑模型；针对逻辑模型可对所学存储结构进行比较、组合、优化，设计出合适存储结构 | 针对考核问题，在时间限定内，可分析得出正确逻辑模型；针对逻辑模型可对所学存储结构进行比较，结合比较结果选择合适的存储结构 | 针对考核问题，在时间限定内，可分析得出正确逻辑模型；可针对逻辑模型选择合适的存储结构。 | 针对考核问题，在时间限定内，可分析得出正确逻辑模型；可针对逻辑模型给出一种基本存储结构。 | 针对考核问题，在时间限定内，可分析得出基本逻辑模型。 | 针对考核问题，在时间限定内，不能进行基本逻辑模型分析、构建。 |
| 目标2 | 针对考核问题，在时间限定内，可选择合适的经典算法或设计时间性能、空间性能较优的算法进行“问题”解决；可对算法进行时间、空间性能分析，并据分析结果进行算法改进 | 针对考核问题，在时间限定内，可选择合适的经典算法或设计满足时间性能、空间性能要求的算法进行“问题”解决；可对算法进行时间、空间性能分析。 | 针对考核问题，在时间限定内，可选择合适的经典算法或设计基本算法解决“问题”；可对算法进行基本的时间、空间性能分析。 | 针对考核问题，在时间限定内，可选择经典算法或设计基本算法解决“问题”。 | 针对考核问题，在时间限定内，可给出解决“问题”基本思路。 | 针对考核问题，在时间限定内，不能给出解决“问题”基本思路。 |

注：每次随堂小测均以百分计，小测成绩70分达到课程目标

**3．期中考试成绩考核与评价标准**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 课程目标 | 评价标准 | | | | | |
| 100~90分 | 89~80分 | 79~70分 | 69~60分 | 59~20分 | 19~0分 |
| 目标1 | 针对考试命题，在时间限定内，可分析得出正确逻辑模型；针对逻辑模型可对所学存储结构进行比较、组合、优化，设计出合适存储结构；步骤清晰、图表规范。 | 针对考试命题，在时间限定内，可分析得出正确逻辑模型；针对逻辑模型可对所学存储结构进行比较，结合比较结果选择合适的存储结构；步骤清晰、图表较规范。 | 针对考试命题，在时间限定内，可分析得出正确逻辑模型；可针对逻辑模型选择合适的存储结构；具有解题步骤、图表较规范。 | 针对考试命题，在时间限定内，可分析得出正确逻辑模型；可针对逻辑模型给出一种基本存储结构；具有基本图表。 | 针对考试命题，在时间限定内，可分析得出基本逻辑模型；具有部分解题分析或图表。 | 针对考核命题，在时间限定内，不能进行基本逻辑模型分析、构建。 |
| 目标2 | 针对考试命题，在时间限定内，可选择合适的经典算法或设计时间性能、空间性能较优的算法进行“问题”解决；可对算法进行时间、空间性能分析，并据分析结果进行算法改进；逻辑呈现清晰，算法描述合理。 | 针对考试命题，在时间限定内，可选择合适的经典算法或设计满足时间性能、空间性能要求的算法进行“问题”解决；可对算法进行时间、空间性能分析；逻辑较清晰，算法描述较合理。 | 针对考试命题，在时间限定内，可选择合适的经典算法或设计基本算法解决“问题”；可对算法进行基本的时间、空间性能分析；逻辑基本清晰，算法描述较合理。 | 针对考试命题，在时间限定内，可选择经典算法或设计基本算法解决“问题”；具有基本逻辑，具有基本算法描述。 | 针对考试命题，在时间限定内，可给出解决“问题”基本思路；具有部分逻辑分析，具有部分算法描述。 | 针对考试命题，在时间限定内，不能给出解决“问题”基本思路。 |
| 目标3 | 针对考试命题，在时间限定内，可综合应用“问题🡺想法🡺算法🡺程序”问题求解过程，结合实际问题，系统构建逻辑模型构建、存储结构选择及优化、算法设计、算法实现、算法分析。 | 针对考试命题，在时间限定内，可应用“问题🡺想法🡺算法🡺程序”问题求解过程，结合实际问题，完成逻辑模型构建、存储结构对比选择、算法设计、算法实现、基本算法分析。 | 针对考试命题，在时间限定内，可基本应用“问题🡺想法🡺算法🡺程序”问题求解过程，结合实际问题，进行逻辑模型构建、选择存储结构、设计并实现算法、进行基本算法分析。 | 针对考试命题，在时间限定内，可基本应用“问题🡺想法🡺算法🡺程序”问题求解过程，结合实际问题，完成逻辑模型构建、给出一种基本存储结构、设计并基本实现算法。 | 针对考试命题，在时间限定内，部分应用问题求解过程，结合实际问题，完成部分逻辑模型构建、设计部分存储结构、实现部分算法。 | 针对考试命题，在时间限定内，不能给出解决方案。 |

注：期中考试成绩以百分计，成绩70分达到课程目标。

**4．实验考核与评价标准**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 课程目标 | 评价标准 | | | | |
| 100~90分 | 89~80分 | 79~70分 | 69~60分 | 59~0分 |
| 目标3 | 能够对给定的实际问题抽象正确的数据模型；可以分析、选择、设计、优化存储结构；针对问题在设计满足时间、空间约束条件的算法基础上可进行进一步算法优化；算法实现正确，实验报告撰写规范；能够对实验结果进行分析。 | 能够对给定的实际问题抽象正确的数据模型；可以分析、选择、设计合适的存储结构；可针对问题设计满足时间、空间约束条件的算法；算法实现正确，实验报告撰写规范；能够对实验结果进行分析并进行算法改进。 | 能够对给定的实际问题抽象数据模型；可分析、选择合适的存储结构；算法设计合理，满足题目约束条件；算法实现正确，实验报告撰写较规范；能够对实验结果进行一定的分析。 | 能够对给定的实际问题抽象数据模型；可实现一种存储结构；算法设计基本合理，基本满足题目约束条件；算法实现基本正确，实验报告撰写基本满足规范。 | 部分完成实验任务，实验报告撰写一般。 |

注：实验成绩以百分计，成绩70分达到课程目标。

**5．期末考试成绩考核与评价标准**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 课程目标 | 评价标准 | | | | | |
| 100~90分 | 89~80分 | 79~70分 | 69~60分 | 59~20分 | 19~0分 |
| 目标1 | 针对考试命题，在时间限定内，可分析得出正确逻辑模型；针对逻辑模型可对所学存储结构进行比较、组合、优化，设计出合适存储结构；步骤清晰、图表规范。 | 针对考试命题，在时间限定内，可分析得出正确逻辑模型；针对逻辑模型可对所学存储结构进行比较，结合比较结果选择合适的存储结构；步骤清晰、图表较规范。 | 针对考试命题，在时间限定内，可分析得出正确逻辑模型；可针对逻辑模型选择合适的存储结构；具有解题步骤、图表较规范。 | 针对考试命题，在时间限定内，可分析得出正确逻辑模型；可针对逻辑模型给出一种基本存储结构；具有基本图表。 | 针对考试命题，在时间限定内，可分析得出基本逻辑模型；具有部分解题分析或图表。 | 针对考核命题，在时间限定内，不能进行基本逻辑模型分析、构建。 |
| 目标2 | 针对考试命题，在时间限定内，可选择合适的经典算法或设计时间性能、空间性能较优的算法进行“问题”解决；可对算法进行时间、空间性能分析，并据分析结果进行算法改进；逻辑呈现清晰，算法描述合理。 | 针对考试命题，在时间限定内，可选择合适的经典算法或设计满足时间性能、空间性能要求的算法进行“问题”解决；可对算法进行时间、空间性能分析；逻辑较清晰，算法描述较合理。 | 针对考试命题，在时间限定内，可选择合适的经典算法或设计基本算法解决“问题”；可对算法进行基本的时间、空间性能分析；逻辑基本清晰，算法描述较合理。 | 针对考试命题，在时间限定内，可选择经典算法或设计基本算法解决“问题”；具有基本逻辑，具有基本算法描述。 | 针对考试命题，在时间限定内，可给出解决“问题”基本思路；具有部分逻辑分析，具有部分算法描述。 | 针对考试命题，在时间限定内，不能给出解决“问题”基本思路。 |
| 目标3 | 针对考试命题，在时间限定内，可综合应用“问题🡺想法🡺算法🡺程序”问题求解过程，结合实际问题，系统构建逻辑模型构建、存储结构选择及优化、算法设计、算法实现、算法分析。 | 针对考试命题，在时间限定内，可应用“问题🡺想法🡺算法🡺程序”问题求解过程，结合实际问题，完成逻辑模型构建、存储结构对比选择、算法设计、算法实现、基本算法分析。 | 针对考试命题，在时间限定内，可基本应用“问题🡺想法🡺算法🡺程序”问题求解过程，结合实际问题，进行逻辑模型构建、选择存储结构、设计并实现算法、进行基本算法分析。 | 针对考试命题，在时间限定内，可基本应用“问题🡺想法🡺算法🡺程序”问题求解过程，结合实际问题，完成逻辑模型构建、给出一种基本存储结构、设计并基本实现算法。 | 针对考试命题，在时间限定内，部分应用问题求解过程，结合实际问题，完成部分逻辑模型构建、设计部分存储结构、实现部分算法。 | 针对考试命题，在时间限定内，不能给出解决方案。 |

注：期末考试成绩以百分计，成绩70分达到课程目标。