

兰州文理学院 2021 年普通专升本招生

“数字媒体技术”专业考试大纲

(试行)

一、考试目的

全面考核普通高等学校高职(专科)应届毕业生数字媒体技术专业核心课程是否达到教学大纲所规定的目标。数字媒体技术专业设置的核心课程主要有:数据结构、操作系统。甘肃省普通高等学校专升本招生数字媒体技术专业的考试,侧重考核数据结构、操作系统课程的学习是否达到了教学大纲所规定的目标。

二、考试范围

主要涵盖专科教学大纲所规定的数据结构、操作系统的内容,并参照本科数字媒体技术专业一年级和二年级的教学内容。重点考核学生数字媒体技术专业基础知识的掌握情况及其应用能力。考试不追求偏题怪题,以基础知识为出题的核心内容。为保证试卷的信度,除论述题带有较强的主观性外,其余试题均以客观性较强的形式为主。

三、考试内容

考试内容以指定教材为主,题型包括五种:单项选择题、多项选择题、判断题、简答题、论述题。

教材:

1. 《实用数据结构基础(第三版)》,陈元春,中国铁道出版社;
2. 《计算机操作系统》,汤小丹,西安电子科技大学出版社,第四版。

四、试题难易度

较容易题	约 30%
中等难度题	约 60%
较难题	约 10%

五、说明

试卷满分为 200 分，2 门课程所占分值为：数据结构和操作系统各占 100 分。考试时间为 180 分钟。

（一）数据结构 部分（100 分）

一、考试目的

全面考核普通高校专科（含高职）应届毕业生《数据结构》课程是否达到教学大纲所规定的目标。数据结构部分主要考核对数据结构中基本概念和原理的掌握情况；要求考生具有抽象思维能力、逻辑推理能力、综合运用所学的知识分析问题和解决问题的能力，以及基本程序设计和编程能力。

二、考试范围

主要涵盖陈元春主编《实用数据结构》（中国铁道出版社 2011 年第三版）中所选定的内容，重点考核数据结构中线性表、栈、队列、树、图、查找、排序等内容的基本概念、原理及其应用能力。考试不追求偏题怪题，以基础知识为出题的核心内容。所有试题都严格选自课本内容。

三、考试内容

考试内容包括七个部分：基础知识、线性表、栈、队列、串、多

维数组、广义表、树、二叉树、图、查找、排序等内容的掌握，以及对相关简单算法的掌握和应用。

第一部分：基础知识，共 5 分。

该部分考察学生对数据结构基本概念和术语，算法的概念、描述方法以及算法分析的理解和掌握情况。

(1) 理解数据、数据元素、数据对象、数据结构和数据类型等概念的定义。

(2) 掌握数据的逻辑结构和存储结构及其种类，算法的重要特征。

(3) 了解算法的时间复杂度和空间复杂度的概念。

第二部分：线性表，共 10 分。

该部分考察学生对线性表的定义、逻辑结构、顺序存储结构和链式存储结构方式，单向链表、循环链表和双向链表的概念和表示，一元多项式的表示及相加等内容的理解和掌握情况。

(1) 了解线性表的定义和线性结构的特点。

(2) 理解线性表的顺序存储和链式存储方式，理解顺序和链式存储方式的优缺点。

(3) 掌握线性顺序表中数据元素的存储位置的计算。

(4) 掌握顺序表、单向链表、循环链表和双向链表的插入、删除等有关操作。

(5) 会用单链表编写插入、删除等有关算法。

第三部分：栈和队列，共 10 分。

该部分考察学生对栈的定义、表示和实现，队列的定义、表示和实现，链队列，循环队列，栈的应用等内容的理解和掌握情况。

(1) 了解栈和队列的定义。

(2) 理解线性表、栈和队列特点及区别，栈对实现递归过程的作用。

(3) 掌握顺序栈、链栈的入栈和出栈操作，顺序队列、链队列的入队和出队操作，循环队列的队空和队满的判断条件。

(4) 会编写入栈和出栈，入队和出队的有关算法。

(5) 了解栈的相关应用，数制转换、表达式求值、子程序调用、递归调用等算法。

第四部分：串、多维数组和广义表，共 10 分。

该部分考察学生对串的有关定义、逻辑结构、静态存储结构、动态存储结构以及基本操作及应用，数组的定义和运算、顺序存储结构，特殊矩阵、稀疏矩阵的定义、压缩存储，广义表的定义、存储结构等内容的理解和掌握情况。

(1) 了解串的有关定义。

(2) 理解串的逻辑结构和定长顺序存储结构、链接存储以及堆分配存储结构。

(3) 掌握串的模式匹配传统方法。

(4) 了解多维数组、特殊矩阵和稀疏矩阵的定义，广义表的概念和链表表示。

(5) 理解特殊矩阵的压缩存储的概念。

(6) 掌握矩阵的压缩存储的有关计算方法。

(7) 掌握广义表的首尾存储法。

第五部分：树和二叉树，共 20 分。

该部分考察学生对树的结构定义和基本操作，二叉树的定义、性质、存储结构、遍历二叉树和线索二叉树，树的存储结构，森林与二叉树的转换，树的遍历，标识符树和哈夫曼树及其编码等内容的理解和掌握情况。

(1) 了解树的定义和二叉树的定义。

(2) 理解二叉树的性质、二叉树的存储结构。

(3) 掌握遍历二叉树的方法、线索二叉树的构造，森林与二叉树的转换，最优二叉树和哈夫曼编码方法。

(4) 会利用二叉树的先根、中根和后根遍历解决有关二叉树的应用问题，会编写与二叉树有关的算法。

第六部分：图，共 15 分。

该部分考察学生对图的定义和术语，图的基本操作，邻接矩阵和邻接表，深度优先搜索和广度优先搜索，无向图的连通分量和生成树、最小生成树，拓扑排序，关键路径等内容的理解和掌握情况。

(1) 了解图的定义和术语，生成树和最小生成树等基本概念。

(2) 理解邻接矩阵中元素的含义和邻接表中结点的含义。

(3) 掌握深度优先搜索和广度优先搜索算法。

(4) 会用 Prim 算法和 Kruskal 算法构造最小生成树。

(5) 会找出图中顶点的拓扑序列，能够利用 floyd 算法和 DijKstra

算法找到顶点间最短路径。

第七部分：排序，共 15 分。

该部分考察学生对直接插入排序、折半插入排序、希尔排序、冒泡排序、快速排序、简单选择排序、堆排序、归并排序、基数排序以及各种内部排序方法的比较等内容的理解和掌握情况。

(1) 理解直接插入排序、折半插入排序、希尔排序、冒泡排序、快速排序、简单选择排序、堆排序、归并排序和基数排序算法的基本思想。

(2) 掌握上述排序算法的稳定性和时间复杂度分析。

(3) 会用上述排序方法写出每趟排序的结果，会编写与直接插入排序和简单选择排序有关的算法。

第八部分：查找，共 15 分。

该部分考察学生对顺序查找、二分查找、分块查找、二叉排序树、平衡二叉树、哈希查找、哈希函数的构造方法和处理冲突的方法等内容的理解和掌握情况。

(1) 了解查找、关键字、平均查找长度等基本概念。

(2) 掌握顺序查找、二分查找、分块查找、二叉排序树、平衡二叉树、哈希查找等基本算法思想及其查找效率。

(3) 掌握哈希函数的构造方法和处理冲突的方法，平衡二叉树的查找、插入和删除操作算法。

(二) 操作系统 部分 (100 分)

一、考试目的

全面考核普通高校专科(含高职)应届毕业生《数据结构》和《操作系统》两门课程是否达到教学大纲所规定的目标,《操作系统》部分主要考核学生对操作系统的基础概念和基本理论的掌握情况。

二、考试范围

主要涵盖汤小丹、梁红兵主编的《计算机操作系统》(第四版)的内容,重点考核学生对操作系统的基本概念、基本原理的掌握情况,考试不追求偏题怪题,以基础知识为出题的核心内容。除综合应用题是考查学生对该门课程知识的整合应用外,其余试题都严格选自课本内容。

三、考试内容

第一章 操作系统引论, 约 10 分。

了解操作系统的定义及作用;掌握操作系统的发展过程;掌握操作系统的分类;重点识记操作系统的基本特性与主要功能;了解 OS 的结构设计。

第二章 进程的描述与控制, 约 10 分。

了解进程的定义和特征;掌握进程的基本状态及转换(三态模型);了解 PCB 的概念及进程的控制;掌握并灵活运用信号量机制解决经典的进程同步问题;了解线程的概念。

第三章 处理机调度与死锁, 约 20 分。

了解处理机调度的三个层次;熟练掌握常用的调度算法(先来先

服务，短作业优先，高响应比优先);了解作业，进程和程序之间的区别和联系;理解并掌握死锁的概念、起因及产生死锁的必要条件;了解处理死锁的四种对策(预防，避免，检测和解除);掌握并灵活应用死锁避免中的银行家算法;

第四章 存储器管理，约 15 分。

了解连续分配的四种存储器管理方式(单一连续、固定分区、动态分区、可重定位);掌握基本的两种离散的存储管理方式(基本分页、基本分段);了解分页和分段的主要区别。

第五章 虚拟存储器，约 10 分。

掌握虚拟存储器的定义;理解并熟练掌握各种页面置换算法(FIFO, LRU, LRU 近似算法)。

第六章 输入输出系统，约 15 分。

了解设备管理的基本功能;掌握四种输入输出控制方式(程序查询方式、中断方式、DMA 方式、通道方式);了解 I/O 软件的层次结构(中断处理、设备驱动、设备独立性、用户层软件);掌握假脱机(SPOOLing)系统的构成;了解缓冲的概念;理解并掌握磁盘调度算法并进行分析(先来先服务、最短寻道时间、扫描算法、循环扫描算法)。

第七章 文件管理，约 10 分。

了解文件系统的基本功能;了解文件的逻辑结构(记录式文件，流式文件)和物理结构(连续结构，串连结构，索引文件);掌握文件的存取方法(顺序存取法，直接存取法，按键存取法);掌握简单的文件目录。

第八章 磁盘存储器的管理，约 5 分。

了解几种常见的文件存储空间的管理方法(空闲表法、空闲链法、位示图法)；了解廉价冗余阵列。

第九章 操作系统接口，约 5 分。

了解两类接口(用户接口、程序接口)；了解系统功能调用。