**《软件技术基础实验》大纲**

**一、实验项目基本信息**

实验项目名称：软件技术基础实验

适用专业：电子信息类（非计算机专业）

学时数： 8

先修课程：《C语言》

考核方式：实验过程、实验报告、实验

**二、实验项目的性质和任务**

1. **实验项目的性质**

本实验是本科生学科拓展课程《软件技术基础》的实验部分，属软件设计的范畴。

1. **实验项目的目的和任务**

通过上机实验，要求学生对常用数据结构的基本概念及其不同的实现方法的理论得到进一步的掌握，并对在不同存储结构上实现不同的运算方式和技巧有所体会。培养学生在软件设计方面的知识基础，同时也提供必要的技能训练。

1. **实验内容和要求**

《软件技术基础实验》是与理论课程《软件技术基础》配套开出的，主要包含两个数据结构部分的基本上机实验和三个课外实验，着重培养同学进行软件设计的基本能力。

**（1）上机实验内容：**

**实验一：顺序表和单链表基本操作上机实验 4学时**

实验目的：理解线性结构的基本概念，掌握两种基本的存储结构：顺序存储结构（顺序表）和链式存储结构（单链表）；用C语言实现在两种存储结构上的对应操作（包括创建、删除插入元素、遍历等），巩固强化C程序设计的基本方法和能力。

 实验内容：

1、顺序表的创建、元素删除、遍历等操作：

有序的一组整数｛1，2，3，4，6｝，设计顺序表并实现以下操作：

A.初始化一个空的顺序表；

B.从键盘依次输入上述数据添加到顺序表中；

C.删除表中的第四个数据元素；

D.显示B、C操作后顺序表中的内容。

2、现有有序的一组整数{1，2，2，3，4，6，6}，设计单链表，分别编写函数实现以下操作：

 A.初始化一个空链表。

 B.依次添加上述一组数据（结点），生成该链表。

 C.统计该链表的长度。

 D.在表中查找数据为3和7的结点，并返回其位置（若找不到返回 -1）。

 E.删除中间重复结点，使链表变为 1-2-3-4-6。

 F.显示经B,E操作后，链表的内容。

**实验二：基本查找排序操作上机实验 4学时**

实验目的：掌握两种基本的查找算法（顺序查找、二分查找）和三种基本排序算法（简单插入排序、简单冒泡排序、简单选择排序）的原理及C语言实现。

实验内容：

现有一顺序表，表中元素分别为{51,38,79,22,91,105,33,52,16,112}，分别编写函数实现以下操作：

A、采用简单插入排序法实现对顺序表的排序，显示每一趟的排序结果；

B、采用简单冒泡排序法实现对顺序表的排序，显示每一趟的排序结果；

C、采用简单选择排序法实现对顺序表的排序，显示每一趟的排序结果；

D、对上述排序后的顺序表采用顺序查找方法，查找关键字为52和关键字为36的元素，分别显示查找结果；

E、对上述排序后的顺序表采用二分查找方法，查找关键字为22的元素，显示查找结果。

**（2）课外实验内容：**

本环节设置了三个相对较复杂的课外实验，旨在进一步巩固学生对数据结构相关知识点的理解和掌握，提高软件开发能力。

**实验三：较复杂的顺序表操作**

实验目的：进一步掌握复杂顺序表的相关操作及编程实现。

实验内容：

图书资料管理问题，设一本书有编号、书名、作者、价格等信息，如下表所示：

|  |  |  |  |
| --- | --- | --- | --- |
| 编号 | 书名 | 作者 | 价格 |
| 1 | Basic | zhang | 15 |
| 2 | VB | Li | 20 |
| 3 | C | Wang | 18 |
| 4 | 　delphi | Zhao | 23 |

　　　　　　设计顺序表并实现以下操作：

　　　　　　　　A、初始化线性表；

　　　　　　　　B、向表中添加一条新记录；

　　　　　　　　C、删除表中的第三条记录；

实验提示：本实验中图书信息的数据类型可定义如下：

　　　　　　　　typedef struct

 　　　　　　　 {

long int number; // 书号

 　　　　　　　　　char name[50]; // 书名

 char author[20]; // 作者

 int price; // 价格

 } bookinfo;

 因此顺序表的定义也应作相应的变化：

 typedef struct

 {

bookinfo data[MAXNUM];

 int num;

 } listtype;

 请同学们思考，插入和删除的函数要做怎样的修改？

**实验四：栈和队列的操作实现**

实验目的：理解栈和队列的基本概念，栈和队列操作的基本方法以及其编程实现。通过本实验的两个项目的编程、调试和运行结果的比较，分析栈和队列的差别。

实验内容：

1、 设计一个容量为4的循环队列，编程实现如下操作，并显示各步骤操作后队列的内容：

A、队列初始化为空；

B、将1、2、3三个数据依次做入队操作；

C、做两次出队操作（1、2出队）；

D、将4、5、6三个数据依次做入队操作；

E、将数据7做入队操作；

2、设计一个容量为4的顺序栈，编程实现如下操作，并显示各步骤操作后栈的内容：

A、栈初始化为空；

B、将1、2、3三个数据依次做入栈操作；

C、做两次出栈操作；

D、将4、5、6三个数据依次做入栈操作；

E、将数据7做入栈操作；

**实验五：查找和排序操作实验**

实验目的：理解在线性表中的各种基本的查找和查找方法的原理，实现方法，适用条

件，掌握各种基本查找和排序方法编程实现。

实验内容：设计一个描述学生成绩信息的线性表，用于存放学生的相关信息，学生成绩表的基本信息如下：

|  |  |  |
| --- | --- | --- |
| 姓名 | 学号 | 成绩 |
| 英语 | 模电 | 高数 | C语言 | 物理 |
| 成一鸣 | 2602302001 | 73 | 82 | 83 | 77 | 92 |
| 傅强之 | 2602302002 | 65 | 76 | 73 | 82 | 84 |
| 郭志川 | 2602302003 | 87 | 86 | 91 | 83 | 81 |
| 姜博文 | 2602302004 | 51 | 78 | 62 | 60 | 80 |
| 巨乐 | 2602302005 | 69 | 68 | 78 | 72 | 79 |
| 李帅 | 2602302006 | 76 | 62 | 58 | 63 | 73 |
| ‥‥‥ | ‥‥‥ | ‥‥ | ‥‥ | ‥‥ | ‥‥ | ‥‥ |

要求编程实现以下功能：

1. 用顺序表存储以上学生成绩表；
2. 添加一个同学的相关信息；
3. 用顺序查找方法查找有某科成绩高于90分或某科成绩不及格的同学信息，并显示相应查找结果；
4. 以英语成绩为排序码，利用简单选择排序方法对学生成绩表进行排序，并显示排序结果；
5. 以高数成绩为排序码，利用简单插入排序方法对学生成绩表进行排序，并显示排序结果；
6. 以C语言成绩为排序码，利用冒泡排序方法对学生成绩表进行排序，并显示排序结果；
7. 再利用二分查找的方法在已经排序后的成绩表中查找C语言成绩为x的学生信息（x为键盘输入分数值）。
8. **实验项目涉及核心知识点：**

对最典型的线性数据结构（线性表）采用不同的存储结构时的基本算法的C语言实现；栈和队列的基本操作C语言实现，查找和排序的操作C语言实现；

**5、重点和难点**

重点在理解数据结构的本质，学会用数据结构中的分析方法来刻画和解决实际问题，并利用C语言为工具来编程验证这些数据结构中的基本算法；

 **6、实验步骤**

A、问题分析和任务的定义

明确问题要求做什么，限制做什么（本步强调做什么，而不是怎么做）。对问题的描述应避开算法和所涉及的数据类型，而是所完成的任务做出明确的回答。如输入数据的类型、值的范围以及输入的形式；输出数据的类型、值的范围及输出的形式；这一步还应该为调试程序准备好测试数据，包括合法的输入数据和非法形式的输入数据。

B、数据类型和系统设计

在设计这一步骤中分为逻辑设计和详细设计两步实现。逻辑设计指 的是，为问题的描述中涉及的操作对象定义相应的数据类型，并按照以数据结构为中心的原则划分模块，定义主模块和各抽象数据类型；详细设计则为定义相应的存储结构并写出各函数的伪码算法。在这个过程中，要综合考虑系统的功能，使得系统结构清晰、合理、简单和易于调试，抽象数据类型的实现尽可能做到数据的封装，基本操作的规格说明尽可能的明确和具体。作为逻辑设计的结果。应写出每个抽象数据类型的定义（包括数据结构的描述和每个基本操作的规格说明），各个主要模块的算法，并画出模块之间的调用关系图。详细设计的结果是对数据结构和基本操作的规格说明做出进一步的求精，写出数据存储结构的类型定义，按照算法书写规范用类C语言写出函数形式的算法框架。

C、编码实现和静态检查；

D、上机准备和上机调试；

E、总结和整理上机实验报告；

7、**使用教材和参考资料**

教材名称及教材性质：

《软件技术基础》，黄迪明，电子科技大学出版社；

**附：实验报告规范**

实验题目

班级 姓名 学号 日期

一、需求分析

1、 程序基本功能

2、 输入输出要求

3、 测试数据

二、概要设计

1. 本程序所用的抽象数据类型的定义；
2. 主程序的流程及各程序模块之间的层次关系。

三、 详细设计

1、 采用c语言定义相关的数据类型；

2、 写出各模块的伪码算法；

3、 画出函数的调用关系图。

四、 调试分析

1、 调试中遇到的问题及对问题的解决方法；

2、 算法的时间复杂度和空间复杂度。

五、 使用说明或测试结果

六、 源程序（带注释）