

《计算机实践》—数据结构 上机实验题和要求

南京航空航天大学
自动化学院
2022 年 10 月

实验 1. 线性表的实验（实验学时：4 学时）

实验目的：

掌握线性表的顺序存储结构和链式存储结构的相关操作实现。

实验 1.1 顺序表的相关运算

1、实验内容

顺序表的定义、创建、插入和删除操作实现。

2、实验要求

- (1) 初始化顺序表，并创建顺序表的初始序列；
- (2) 在指定位置实现顺序表的插入操作，并输出插入操作后的序列；
- (3) 在指定位置实现顺序表的删除操作，并输出删除操作后的序列。

3、输入输出要求

- (1) 输入数据：建立输入处理，输入顺序表里的数据（逻辑顺序预先确定，数据类型为字符型或整型，数据值自行设置），插入和删除的位置要求从键盘输入。
- (2) 输出形式：分别输出顺序表的初始序列以及进行插入、删除操作以后的正确序列。

实验 1.2 单链表的相关运算

1、实验内容

单链表的定义、创建及相关操作实现。

2、实验问题描述

求先后输入的两个集合 A、B 的差集（A-B）和并集（A+B）。

说明：已知两个集合 A 和 B（每个集合内部元素各不相同），两个集合的差集（A-B）中包含所有属于集合 A 而不属于集合 B 的元素；两个集合的并集（A+B）中包含所有属于集合 A 和属于集合 B 的元素，且相同数据只存储一个。

3、实验要求

- (1) 集合 A、B 采用单链表的存储结构；
- (2) 输出 A、B 原始序列、差集和并集。

4、输入输出要求

- (1) 输入数据：建立输入处理，输入单链表 A 和 B 里的数据（逻辑顺序预先确定，数据类型为字符型或整型，数据值自行设置）。
- (2) 输出形式：分别输出单链表 A 和 B 的原始序列、差集（A-B）和并集（A+B）。

5、提示：求差（A-B）：

步骤：1) 首先创建 A 和 B 两个单链表；

2) 然后对集合 B 中的每个元素 x，在 A 中查找，若存在和 x 相同的元素，则从 A 链表中删除；

3) 输出 A-B，进行验证。

6、测试用例

序号	输入		输出	
	集合 A	集合 B	差(A-B)	并(A+B)
1	1 2 3 4 5 6 7	1 3 5 7 9 11	2 4 6	1 2 3 4 5 6 7 9 11
2	1 3 5 7 9 11	1 2 3 4 5 6 7	9 11	1 2 3 4 5 6 7 9 11

实验 1.3 线性表元素的区间删除（选做）

1. 实验内容

创建一个线性表，然后设计一个函数删除所有值大于 minD 并且小于 maxD 的数据元素。删除后表中剩余数据元素保持原有存储结构，并且相对位置保持不变。

2. 实验要求

(1) 函数接口说明：

Delete(线性表类型名 *L, ElementType minD, ElementType maxD)，其中线性表数据元素设定为整数。

(2) 使用顺序存储实现实验内容；

(3) 使用链式存储实现实验内容。

3. 测试用例

序号	输入				输出		说明
	元素个数 n	L	minD	maxD	元素个数 m	L	
1	10	4 -8 2 12 1 5 9 3 3 10	0	4	6	4 -8 12 5 9 10	删除中间元素
2	6	23 21 46 9 90 12	9	46	3	46 9 90	删除头尾
3	6	1 2 3 4 5 6	0	100	0	空	全部删除
4	1	256	10	200	1	256	无删除

4. 输入输出要求

输入数据：通过合适的方式输入元素个数 n 和元素的值，创建线性表；然后输入 minD

和 $\max D$ 的值 ($\min D \leq \max D$);

输出数据: 首先输出线性表元素个数 n , 然后输出初始序列; 换行后输出区间删除操作以后的元素个数 m 和正确序列。

5. 选作内容

尝试分析两种存储结构中, 区间删除操作的时间复杂度和空间复杂度。

实验 1.4 约瑟夫斯问题求解 (选做)

1. 问题描述

约瑟夫斯 (Josephus) 问题的一种描述是: 编号为 $1, 2, \dots, n$ 的 n 个人按顺时针方向围坐一圈, 每人持有一个密码 (正整数)。一开始任选一个正整数作为报数上限值 m , 从第一个人开始按顺时针方向自 1 开始报数, 报到 m 时停止报数。报 m 的人出列, 将他的密码作为新的 m 值, 从他在顺时针方向下一个人开始重新从 1 报数, 如此下去, 直至所有的人全部出列为止。试设计一个程序, 按出列顺序印出各人编号。

2. 基本要求

利用单向循环链表存储结构模拟此过程, 按照出列的顺序印出各人的编号。

3. 测试数据

$n=7$, 7 个人的密码依次为: 3, 1, 7, 2, 4, 8, 4。 m 初值为 6 (正确的出列顺序应为 6, 1, 4, 7, 2, 3, 5)。

4. 提示

程序运行后, 首先要求用户指定初始报数上限 m , 然后读取个人的密码。可设 $n \leq 30$ 。注意链表中空表和非空表的界限。

5. 输入输出

输入数据: 建立输入处理, 输入 n 输入以及每个人的密码; m 的初值。

输出形式: 建立一个输出函数, 输出正确的序列。

6. 选作内容

添加采用顺序存储结构实现问题求解的模块。