

山东大学计算科学与技术学院

《数据结构与算法》实验指导书

一、实验要求

- 一、 采用良好的编程风格；关键操作要有注释。
- 二、 程序能够运行，显示运行结果。

二、 开发工具

Microsoft Visual C++

Eclipse IDE For C++

实验室安装：Dev C++; visual studio 等

三、 实验题目

实验一 递归练习

一、实验目的

- 1、熟悉开发工具的使用。
- 2、掌握递归的实现思想。

二、 实验内容

1、题目描述：

现有一个有 n 个元素的序列 $a=[a_1, a_2, \dots, a_n]$ ，定义这个序列的价值为

$\sum_{i=1}^n i \times a_i$ 。空序列的价值为 0。先给你一个长度为 n 的序列 a ，求 a 中所有子集价值的异或和，要求子集中元素的相对位置保持不变。

异或和：位运算的一种。如果 a 、 b 两个值不相同，则异或结果为 1；如果 a 、 b 两个值相同，异或结果为 0。

输入输出格式：

输入： 第一行，一个整数 n

接下来一行有 n 个非负整数： a_1, a_2, \dots, a_n

输出： 一个整数，表示所有子集价值的异或和。

2、题目描述：

现有一个有 n 个元素的序列 $a=[a_1, a_2, \dots, a_n]$ ，定义其价值为 $\sum_{i=1}^n a_i \oplus i$

给出这样一个序列，求其所有排列的价值 v_i 或 $V_1 | V_2 | V_3 | \dots | V_{n-1} | V_n$

其中 $|$ 为位运算或操作， \oplus 为位运算异或操作。

输入输出格式：

输入： 输入的第一行是一个整数 n ($2 \leq n \leq 10$)，表示需排列的数的个数。

接下来一行是 n 个整数，数的范围是 0 到 100000，每两个相邻数据间用一个空格分隔。

输出：

一个整数，代表所有排列价值的或。

实验二 排序算法

一、实验目的

掌握各种简单排序算法。

二、实验内容

1、题目描述：

用任意一种排序方式给出 n 个整数按升序排序后的结果，满足以下要求：

1. 不得使用与实验相关的 STL；

- 2.需使用类模版(template<class T>);
- 3.需定义排序类, 封装各排序方法;
- 4.排序数据需使用动态数组存储;
- 5.排序类需提供以下操作: 名次排序、及时终止的选择排序、及时终止的冒泡排序、插入排序。

输入输出格式:

输入: 输入的第一行是一个整数 $n(1 \leq n \leq 1000)$, 表示需排序的数的个数。
接下来一行是 n 个整数, 数的范围是 0 到 1000, 每两个相邻数据间用一个空格分隔。

输出: 一行排好序的序列。

实验三 数组描述线性表

一、实验目的

- 1、掌握线性表结构、数组描述方法(顺序存储结构)、数组描述线性表的实现。
- 2、掌握线性表应用。

二、实验内容

1、题目描述:

设通讯录中每一个联系人的内容有: 姓名、电话号码、班级、宿舍。由标准输入读入联系人信息, 使用线性表中操作实现通讯录管理功能, 包括: 插入、删除、编辑、查找(按姓名查找); 键盘输入一班级, 输出通讯录中该班级中所有人的信息。

每个操作的第一个数为操作数(插入-0, 删除-1, 编辑-2, 查找-3, 输出一个班所有人员信息-4), 具体格式如下:

- 0 姓名 电话 班级 宿舍 插入一条记录
- 1 姓名 根据姓名删除一条记录
- 2 姓名 编辑项目 项目新值 根据姓名编辑一条记录(编辑项目为 1 到 3 的整数, 1 代表编辑电话, 2 代表编辑班级, 3 代表编辑宿舍)

- **3 姓名** 根据姓名查找，找到输出 1，未找到输出 0
- **4 班级** 输出该班级的所有成员的宿舍号的异或值

其中查找操作当找到相应的人时输出 1，未找到输出 0。输出一个班级的人员信息时输出所有成员的宿舍号的异或值。输入数据保证合法。

输入输出格式：

输入：

第一行一个 $n(1 \leq n \leq 20000)$ ，代表接下来操作的数目。接下来 n 行代表各项操作。

输出：

当遇到查找和输出一个班所有人员信息操作时输出。

实验四 链式描述线性表

一、实验目的

- 1、掌握线性表结构、链式描述方法（链式存储结构）、链表的实现。
- 2、掌握链表迭代器的实现与应用。

二、实验内容

1、题目描述：

要求封装链表类，链表迭代器类；

链表类需提供操作：在指定位置插入元素，删除指定元素，搜索链表中是否有指定元素，原地逆置链表，输出链表；

不得使用与链表实现相关的 STL。

输入输出格式：

输入： 第一行两个整数 N 和 Q 。

第二行 N 个整数，作为节点的元素值，创建链表。

接下来 Q 行，执行各个操作，具体格式如下：

插入操作：1 idx val，在链表的 idx 位置插入元素 val；

删除操作：2 val，删除链表中的 val 元素。若链表中存在多个该元素，仅删除第一个。若该元素不存在，输出 -1；

逆置操作：3，原地逆置链表；

查询操作：4 val，查询链表中的 val 元素，并输出其索引。若链表中存在多个该元素，仅输出第一个的索引。若不存在该元素，输出 -1；

输出操作：5，使用链表迭代器，输出当前链表索引与元素的异或和。

$$f(chain) = \sum_{i=0}^{n-1} i \oplus chain[i], n = len(chain);$$

2、题目描述：

要求使用题目一中实现的链表类，迭代器类完成本题；

不得使用与题目实现相关的 STL；

给定两组整数序列，你需要分别创建两个有序链表，使用链表迭代器实现链表的合并，并分别输出这三个有序链表的索引与元素的异或和。

注：给定序列是无序的，你需要首先得到一个有序的链表。

输入输出格式：

输入：

第一行两个整数 N 和 M；

第二行 N 个整数，代表第一组整数序列；

第三行 M 个整数，代表第二组整数序列。

输出：

三行整数。分别代表第一组数、第二组数对应的有序链表与合并后有序链表的索引与元素的异或和。

实验五 数组和矩阵

一、实验目的

掌握稀疏矩阵结构的描述及操作的实现。

二、实验内容

1、题目描述：

创建稀疏矩阵类（参照课本 `MatrixTerm` 三元组定义），采用行主顺序把稀疏矩阵非 0 元素映射到一维数组中，实现操作：两个稀疏矩阵相加、两个稀疏矩阵相乘、稀疏矩阵的转置、输出矩阵。

重置矩阵：操作 1，即重置矩阵 P 的尺寸为 n 行 m 列,且随后按行优先顺序输入矩阵 P 的各个元素。

矩阵乘法：操作 2， t 行非零元素已按行优先顺序给出，矩阵中非零元素的表示为 $x\ y\ v$ ，其中 x 表示行序号， y 表示列序号， v 表示非零元素值，行列序号从 1 开始。设输入的矩阵为 Q ，若 $P \times Q$ 运算合法,则将 $P \times Q$ 的结果矩阵赋给 P ,若不合法,则将 Q 赋给 P ，同时输出-1。

矩阵加法：操作 3， t 行非零元素已按行优先顺序给出，矩阵中非零元素的表示为 $x\ y\ v$ ，其中 x 表示行序号， y 表示列序号， v 表示非零元素值，行列序号从 1 开始。设输入的矩阵为 Q ,若 $P+Q$ 运算合法,则将 $P+Q$ 的结果矩阵赋给 P ,若不合法,则将 Q 赋给 P ,同时输出 -1。

输出操作：操作 4，设当前矩阵 P 的尺寸为 n 行 m 列,第一行输出矩阵 P 的行数和列数，随后 n 行按行优先顺序输出矩阵 P ,每行 m 个数字，来表示当前的矩阵内容，每行数字之间用空格分隔。

转置操作：操作 5，设当前矩阵 P 的尺寸为 n 行 m 列，将其转置为 m 行 n 列的矩阵，无需输出。

输入输出格式：

输入：

第一行一个 w 代表操作个数，接下来若干行是各个操作，其中保证第一个操作一定为重置矩阵。

输出：

当执行操作 4 时，输出矩阵 P ；当执行操作 2 或 3 时，若对应运算不合法，则输出-1。

实验六 栈

一、实验目的

- 1、掌握栈结构的定义与实现；

2、掌握栈结构的使用。

二、实验内容

1、题目描述：

创建栈类，采用数组描述；计算数学表达式的值。输入数学表达式，输出表达式的计算结果。数学表达式由单个数字和运算符“+”、“-”、“*”、“/”、“(”、“)”构成，例如 $2+3*(4+5)-6/4$ 。

输入输出格式：

输入：

第一行一个整数 $n(1 \leq n \leq 100)$ ，代表表达式的个数。

接下来 n 行，每行一个表达式，保证表达式内的数字为单个整数，表达式内各运算符和数字间没有空格，且表达式的长度不超过 2000。

输出：

每行表达式输出一个浮点数，要求保留两位小数，保证输入表达式合法。

实验七 队列

一、实验目的

1、掌握队列结构的定义与实现；

2、掌握队列结构的使用。

二、实验内容

1、题目描述：

首先创建队列类，采用数组描述；实现卡片游戏，假设桌上有一叠扑克牌，依次编号为 $1-n$ （从最上面开始）。当至少还有两张的时候，可以进行操作：把第一张牌扔掉，然后把新的第一张放到整叠牌的最后。输入 n ，输出最后剩下的牌。

输入输出格式：

输入：

一个整数 n ，代表一开始卡片的总数。

输出：

最后一张卡片的值。

实验八 散列表

一、实验目的

- 1、掌握散列表结构的定义和实现。
- 2、掌握散列表结构的应用。

二、实验内容

1、题目描述:

给定散列函数的除数 D 和操作数 m ，输出每次操作后的状态。

有以下三种操作：

插入 x ，若散列表已存在 x ，输出 “Existed”，否则插入 x 到散列表中，输出所在的下标。

查询 x ，若散列表不含有 x ，输出 “-1”，否则输出 x 对应下标。

删除 x ，若散列表不含有 x ，输出 “Not Found”，否则输出删除 x 过程中移动元素的个数。

输入输出格式：

输入：

第一行两个整数 D ， m 。分别代表散列函数的除数 D 和操作数 m 。

接下来 m 行，每行两个整数 opt 和 x ，分别代表操作类型和操作数。

若 opt 为 0，代表插入 x ；

若 opt 为 1，代表查询 x ；

若 opt 为 2，代表删除 x 。

输出：

按需输出。

2、题目描述:

给定散列函数的除数 D 和操作数 m ，输出每次操作后的状态。

有以下三种操作：

插入 x ，若散列表已存在 x ，输出 "Existed"；

查询 x ，若散列表不含有 x ，输出 "Not Found"，否则输出 x 所在的链表

长度;

删除 x , 若散列表不含有 x , 输出 "Delete Failed", 否则输出 x 所在链表

删除 x 后的长度;

输入输出格式:

输入:

第一行两个整数 $D(1 \leq D \leq 3000)$ 和 $m(1 \leq m \leq 3000)$, 其中 D 为散列函数的除数, m 为操作数。

接下来的 m 行, 每行两个整数 opt 和 x , 分别代表操作类型和操作数。

若 opt 为 0, 则代表向散列表中插入 x ;

若 opt 为 1, 代表查询散列表中 x 是否存在;

若 opt 为 2, (如果散列表中含有 x), 删除 x 。

输出:

按需输出。

实验九 二叉树操作

一、实验目的

掌握二叉树的基本概念, 链表描述方法; 二叉树操作的实现。

二、实验内容

1、题目描述:

创建二叉树类。二叉树的存储结构使用链表。提供操作: 前序遍历、中序遍历、后序遍历、层次遍历、计算二叉树结点数目、计算二叉树高度。

输入输出格式:

输入:

第一行为一个数字 $n(10 \leq n \leq 100000)$, 表示有这棵树有 n 个节点, 编号为 $1 \sim n$ 。之后 n 行每行两个数字, 第 i 行的两个数字 a, b 表示编号为 i 的节点的左孩子节点为 a , 右孩子节点为 b , -1 表示该位置没有节点。保证数据有效, 根节点为 1。

输出:

第一行， n 个数字，表示该树的层次遍历。

第二行， n 个数字，第 i 个数字表示以 i 节点为根的子树的节点数目。

第三行， n 个数字，第 i 个数字表示以 i 节点为根的子树的高度。

2、题目描述：

接收二叉树前序序列和中序序列(各元素各不相同)，输出该二叉树的后序序列。

输入输出格式：

输入：

第一行为数字 n ；

第二行有 n 个数字，表示二叉树的前序遍历；

第三行有 n 个数字，表示二叉树的中序遍历。

输出：

输出一行，表示该二叉树的后序遍历序列。

实验十 堆及其应用

一、实验目的

- 1、掌握堆结构的定义、描述方法、操作定义及实现。
- 2、掌握堆结构的应用。

二、实验内容

1、题目描述：

创建最小堆类，使用数组作为存储结构，提供操作：插入、删除、初始化、排序。保证第一个操作是建堆操作，接下来是对堆的插入和删除操作，删除和插入都在建好的堆上进行操作。

输入输出格式：

输入：

第一行一个数 n ($n \leq 5000$)，代表堆的大小；第二行 n 个数，代表堆的各个元素；第三行一个数 m ($m \leq 1000$)，代表接下来共 m 个操作。接下来 m 行，分别代表各个操作。下面是各个操作的格式：

插入操作：1 num；

删除操作：2；

排序操作：第一行两个数 3 和 n ，3 代表是排序操作， n 代表待排序的数的数目，接下来一行 n 个数是待排序数。

保证排序操作只出现一次且一定是最后一个操作。

输出：

第一行建堆操作输出建好堆后的堆顶的元素。接下来 m 个操作，若是插入和删除操作。每行输出执行操作后堆顶的元素的值；若是排序操作，输出一行按升序排序好的结果，每个元素间用空格分隔。

2、题目描述：哈夫曼编码。

输入输出格式：

输入：

一串小写字母组成的字符串（不超过 1000000）。

输出：

输出这个字符串通过 Huffman 编码后的长度。

实验十一 搜索树

一、实验目的

掌握二叉搜索树结构的定义、描述方法、操作实现。

三、实验内容

1、题目描述：

创建带索引的二叉搜索树类。存储结构使用链表，提供操作:插入、删除、按名次删除、查找、按名次查找、升序输出所有元素。

输入输出格式：

输入：

输入第一行一个数字 m ($m \leq 1000000$)，表示有 m 个操作。

接下来 m 行，每一行有两个数字 a , b ;

当输入的第二个数字 a 为 0 时，输入的第三个数字 b 表示向搜索树中插入 b ;

当输入的第二个数字 a 为 1 时，输入的第三个数字 b 表示向搜索树

中查找 b ;

当输入的第一个数字 a 为 2 时, 输入的第二个数字 b 表示向搜索树中删除 b ;

当输入的第一个数字 a 为 3 时, 输入的第二个数字 b 表示查找搜索树中名次为 b 的元素;

当输入的第一个数字 a 为 4 时, 输入的第二个数字 b 表示删除搜索树中名次为 b 的元素;

输出:

对于输入中的每一种操作, 输出执行操作的过程中依次比较的元素值的异或值。

实验十二 图 (4 学时)

一、实验目的

- 1、掌握图的基本概念, 图的描述方法; 图上的操作方法实现。
- 2、掌握图结构的应用。

二、实验内容

1、题目描述:

创建无向图类, 存储结构使用邻接链表, 提供操作: 插入一条边, 删除一条边, BFS, DFS。

输入输出格式:

输入:

第一行四个整数 n, m, s, t 。 n ($10 \leq n \leq 100000$) 代表图中点的个数, m ($10 \leq m \leq 200000$) 代表接下来共有 m 个操作, s 代表起始点, t 代表终点。

接下来 m 行, 每行代表一次插入或删除边的操作, 操作格式为:

0 $u v$ 在点 u 和 v 之间增加一条边;

1 $u v$ 删除点 u 和 v 之间的边。

输出:

第一行输出图中有多少个连通分量;

第二行输出所有连通子图中最小点的编号（升序），编号间用空格分隔；

第三行输出从 s 点开始的 dfs 序列长度；

第四行输出从 s 点开始的字典序最小的 dfs 序列；

第五行输出从 t 点开始的 bfs 序列的长度；

第六行输出从 t 点开始字典序最小的 bfs 序列；

第七行输出从 s 点到 t 点的最短路径，若是不存在路径则输出-1。