

实验一 链表的实现与应用

实验目的

1. 熟悉 C/C++ 集成开发环境，能够利用 C/C++ 语言实现基本算法
2. 掌握链表的存储结构，实现链表的基本操作
3. 合理设计运用数据结构，能够编写高效率算法解决应用问题

实验内容

题 1：示例程序调试

根据示例程序理解单链表的数据类型定义和基本操作的实现，熟练在集成开发环境中的编辑、编译、运行和调试方法。示例程序包含头文件 linklist.h，实现文件 Linklist.cpp，以及主程序 main.cpp，其中可能存在若干错误，请尝试排除，并编译运行得到以下正确结果：

```
1) 初始化单链表L
输出单链表L:

2) 顺序插入链表元素a,b,c,d,e
3) 输出单链表L:
a b c d e
4) 单链表L长度=5
5) 单链表L非空
6) 单链表L中第3个元素是: c
7) 元素'b'在单链表中位序是: 2
8) 在单链表L中第4位置插入'f'。
插入成功。
插入后单链表L:
a b c f d e
9) 删除单链表L中第3个元素。
删除成功。
删除后单链表L:
a b f d e
10) 销毁单链表L
输出单链表L:
```

图 1-1 题 1 正确运行结果

题 2：约瑟夫问题求解

约瑟夫 (Josephus) 问题的一种描述是：编号为 1, 2, ..., 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，允许输入 n 以及每个人的密码和 m 的初值，并按出列顺序打印出各人编号。

下图给出当 $n=7$ ， $m=20$ ，个人密码分别为 {3, 1, 7, 2, 4, 8, 4} 时正确的输出序列：

```
输入约瑟夫环的长度：7
请输入m的初始值 m:20
请输入每个人的密码：
3 1 7 2 4 8 4
6
1
4
7
2
3
5
```

图 1-2 题 2 示例运行结果

实验要求

1. 认真阅读并在 5 学时内完成本次实验的所有内容
2. 灵活运用课程所学数据结构，提前做好问题分析与数据建模，完成详细代码设计和上机调试
3. 每次上机结束后请自行使用 U 盘保存好源程序和可执行文件，以备工作接续和检查之需
4. 建议拍照保留程序运行的重要过程和结果，以备撰写实验报告时使用