

实验 1：链表

一、实验要求

- (1) 独立完成实验
- (2) 撰写实验报告
- (3) 掌握链表的基本操作。
- (4) 通过实验加深理解数据结构在实际问题中的应用。

二、实验环境

硬件环境

处理器：13th Gen Intel(R) Core(TM) i7-13700H
机带 RAM：16.0 GB (15.6 GB 可用)
系统类型：64 位操作系统, 基于 x64 的处理器

软件环境

VSCode 版本：1.85.2
编译器：gcc version 8.1.0 (x86_64-posix-sjlj-rev0)

三、实验步骤及思路

(1) 题目分析

实验 1.1

题目要求判断链表 B 是否为链表 A 的连续子序列。这要求我们在遍历链表 A 的过程中，尝试匹配链表 B 的序列。如果能在链表 A 中找到与链表 B 完全匹配连续序列，则说明链表 B 是链表 A 的连续子序列。

实验 1.2

循环双链表是一种特殊的双向链表，其中每个节点都包含指向前一个节点和后一个节点的指针，且首尾相连形成环状。本实验要求实现循环双链表的基本运算，并设计算法判断其是否对称。

(2) 实验具体步骤（思路）

实验 1.1

- 1.输入两个字符串 a 和 b，分别表示链表 A 和链表 B 的内容。
- 2.调用 CreateLink 函数创建两个链表 A 和 B。
- 3.初始化两个指针 t1 和 t2，用于指向比较的内容。使用循环遍历链表 A，当 t1 指向的内容等于 B 的开头时，t2 指向 B 的下一个内容，继续比较。内容不匹配时，t2 指回 B 的开头，等待在 A 中找到下一个等于 B 的开头的内容。
- 4.如果 t2 为空，说明在 A 中找完了 B 的序列，输出“yes”并退出程序。如果遍历完链表 A 仍未找到匹配的序列，输出“no”。

实验 1.2

定义循环双链表结构：

定义一个结构体 Link，包含字符类型的元素 a，以及指向下一个节点和前一个节点的指针 next 和 front。

实现基本运算：

InitLink：初始化循环双链表，创建一个带头节点的空链表，并将其 next 和 front 指针都指向自身，形成循环。

print：输出循环双链表的内容，从头节点开始遍历并输出每个节点的元素。

InsertLink：创建新节点，并更新头节点的 front 指针以及新节点的前一个节点的 next 指针，实现尾插法插入。

getlenth：计算并返回循环双链表的长度。

ifempty：判断循环双链表是否为空，并返回相应的结果。

实现对称判断算法：

设计 ifSymmetry 函数，通过同时从头尾两个方向遍历循环双链表，比较对应位置的元素是否相等，从而判断链表是否对称。

四、实验结果及分析

运行结果（运行截图）

实验 1. 1

```
问题  输出  终端  端口  注释  调试控制台
Active code page: 65001
PS E:\program\structure> cd "e:\program\structure\" ; if ($?) { g++ 1-1.cpp -o 1-1 } ; if ($?) { .\1-1 }
2023302181251
0218
yes
PS E:\program\structure> cd "e:\program\structure\" ; if ($?) { g++ 1-1.cpp -o 1-1 } ; if ($?) { .\1-1 }
2023302181251
21811
no
PS E:\program\structure> 
```

实验 2. 1

```
问题  输出  终端  端口  注释  调试控制台
2023302181251
2 0 2 3 3 0 2 1 8 1 2 5 1
a
lenth:1 ifempty:0
a b
lenth:2 ifempty:0
a b c
lenth:3 ifempty:0
lenth:0 ifempty:1
12321
1
123321
1
12312
0
PS E:\program\structure> 
```

分析

实验 1. 1

当链表 B 是链表 A 的连续子序列时，程序能够正确输出“yes”。

当链表 B 不是链表 A 的连续子序列时，程序能够正确输出“no”。

程序的时间复杂度为 $O(n)$ ，主要取决于链表 A 的长度，最坏情况下需要遍历整个链表 A。

实验 2. 1

第一行输入学号，第二行以空格为分隔输出了链表内容。

接下来 6 行是逐个插入 a, b, c 的时链表内容, 链表长度, 及是否为空 (其中 ifempty 为 0 表示非空, 1 表示空)。

接下来时创建并初始化一个空链表, 链表长度为 0, ifempty 为 1 表示空。

最后 6 行为输入一串字符串, 输出是否对称 (其中 0 表示非对称, 1 表示对称)。

根据实验结果, 程序能够正确初始化循环双链表, 实现尾插法插入元素, 并正确输出链表内容、长度和是否为空的信息。同时, 对称判断算法也能够正确判断循环双链表是否对称。

五、总结

实验完成情况

本实验成功实现了判断链表 B 是否为链表 A 的连续子序列的功能和循环双链表的基本运算和对称判断算法。代码能够正确执行各项操作, 并输出预期结果。

实验体会

通过本次实验, 我深入理解了链表的基本操作和遍历方法, 体会到了如何运用数据结构并设计算法来解决问题。同时, 我也认识到了在编程过程中需要仔细思考, 注意细节, 避免出现逻辑错误。在未来的学习和实践中, 我将更加注重代码的质量和可读性, 不断提高自己的编程能力。