

1. 题目分析

明程序设计的任务，强调的是程序要做什么，此外列出各成员分工

程序设计任务：解决约瑟夫问题——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 开始顺序报数，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，从他在顺时针方向上的下一个人开始重新从 1 报数，如此下去，直至所有人全部出列为止。最后输出列顺序。

成员分工：祝子贤：题意分析，程序的调试和改进，填写实验报告

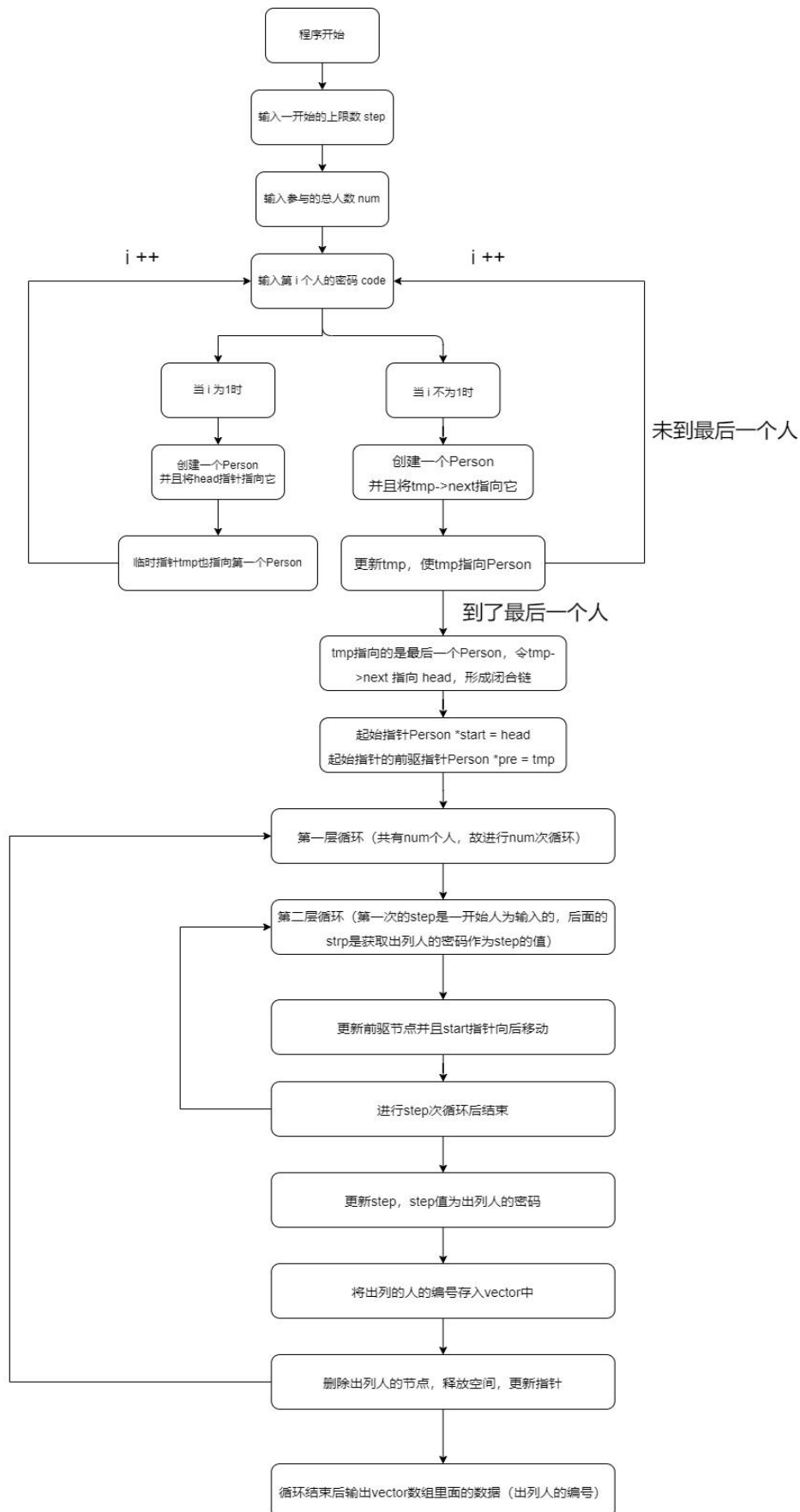
陈欣欣：程序的设计和实现，填写实验报告

2. 数据结构设计

说明程序用到的数据结构的定义，主程序的流程及各模块之间的层次关系

数据结构：单向循环链表。定义了一个 Person 的类（包含了密码（code）、编号（No）、下一个人的指针 $\text{Person}^* \text{next}$ 、以及一个构造函数 $\text{Person}(\text{int } c, \text{int } N)$ 其中 c 表示当前人的 code， N 表示当前人的编号）

主程序流程图（压缩包内有更清晰的 png 和 wml 文件）：



3. 程序设计

实现概要设计中的数据类型，对主程序、模块及主要操作写出伪代码，画出函数的调用关系

数据类型: vector 数组、指针、int

主程序操作伪代码:

```
JosephusProblem主程序的思路以及伪代码

//人类
class Person {
public:
    int code;
    int No;
    Person *next = nullptr;
    Person(int c, int N):code(c), No(N) {
    }
};

//主函数
int main(){
    int step;//下一次要走的步数
    int num;//一共有几个人
    scanf("%d%d",&step,&num);//输入step和num
    Person *head, *tmp;//head为循环单链表的表头，tep为临时变量但最后存放的是最后一个人的位子
    //用两层循环来构造循环单向链表
    for(.....){ //第一层循环
        for(.....){ //第二层循环
            ....
        }
        .....
    }
    tmp->next = head; //将尾衔接到头上
    vecotor<int> NoList//创建一个vector数组用于存放出列人的编号
    //用两层循环来存出列人的编号
    for(.....){
        for(.....){
        }
        step = start -> code // 读取密码作为下一个step值
        push//将start -> No 存入vector数组里
        delete/free/updata //删除节点、释放删除节点的空间、更新指针和变量
    }
    //用一层循环来输出结果
    for(.....){
        cout << i << ' ';
    }
    return 0;
}
```

4. 调试分析

程序复杂度分析：空间复杂度： $O(n)$ ；时间复杂度： $O(m+n+\sum \text{code})$

心得体会：使用链表要注意不能使用空指针，不能使用指向未知内存的指针，以及用完指针后要及时释放。否则就会碰到一些莫名其妙的错误。

5. 测试结果

列出测试结果，包括输入和输出

图一：上限值为 20、人数为 7、每人的密码分别为 3、1、7、2、4、8、4

```
请输入报数上限值20
请输入总共有多少个人7
请输入第1个人的密码3
请输入第2个人的密码1
请输入第3个人的密码7
请输入第4个人的密码2
请输入第5个人的密码4
请输入第6个人的密码8
请输入第7个人的密码4
6 1 4 7 2 3 5
-----
Process exited after 15.08 seconds with return value 0
请按任意键继续. . . |
```

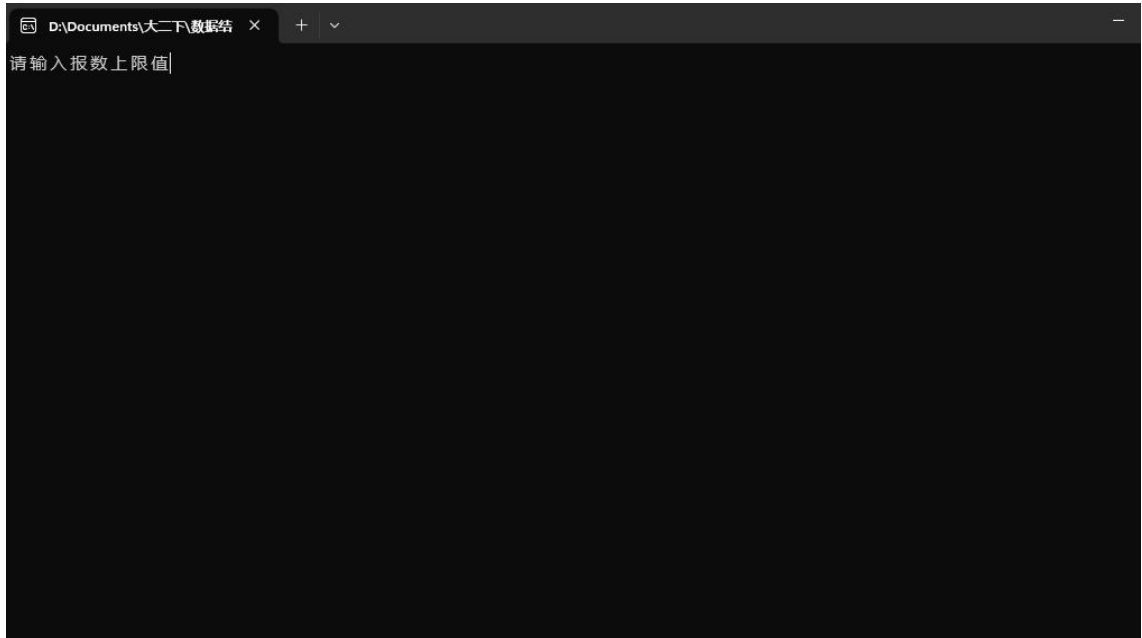
图二：上限值为 30、人数为 7、每人的密码分别为 3、1、7、2、4、8、4

```
请输入报数上限值30
请输入总共有多少个人7
请输入第1个人的密码3
请输入第2个人的密码1
请输入第3个人的密码7
请输入第4个人的密码2
请输入第5个人的密码4
请输入第6个人的密码8
请输入第7个人的密码4
2 3 5 4 7 6 1
-----
Process exited after 13.35 seconds with return value 0
请按任意键继续. . . |
```

6. 用户使用说明

给出主界面及主要功能界面

主界面:



使用说明:

输入上限值后回车 -> 输入人数后回车 -> 输入每个人的密码后回车 -> 全部密码输入完毕后自动打印出列人员的编号。

7. 附录

源程序文件清单:

JosephusProblem.cpp //主程序