项目说明文档——约瑟夫生死游戏

姓名：吴桐欣
学号：1652677
同济大学 软件学院 软件工程专业

目录

1.项目概述

1.1 项目简介

约瑟夫生者死者游戏的大意是：30 个旅客同乘一条船，因为严重超载，加上风高浪大危险万分；因此船长告诉乘客，只有将全船一半的旅客投入海中，其余人才能幸免于难。无奈，大家只得统一这种方法，并议定 30 个人围成一圈，由第一个人开始，依次报数，数到第 9 人，便将他投入大海中，然后从他的下一个人数起，数到第 9 人，再将他投入大海，如此循环，直到剩下 15 个乘客为止。问哪些位置是将被扔下大海的位置。

本游戏的数学建模如下：假如 N 个旅客排成一个环形，依次顺序编号 1, 2, …, N。从某个指定的第 S 号开始。沿环计数，每数到第 M 个人就让器出列，且从下一个人开始重新计数，继续进行下去。这个过程一直进行到剩下 K 个旅客为止。

1.1.1 功能分析

本项目要求实现功能

（1） 输出离开旅客的序号；

（2） 输出剩余旅客的序号。

1.2 文件目录

（1）P02_1652677_吴桐欣_说明文档.docx（本文档）

（2）P02_1652677_吴桐欣.exe（可执行文件）

（3）P02_1652677_吴桐欣.cpp（源文件）

1.3 操作指南

运行程序后，将依次获得程序提示

“总人数 n：

开始位置 s：

循环数 m：

剩余人数 k：”

用户输入按顺序输入 4 个数字，即可获得结果

1.4 注意事项

（1）用户不得输入除数字以外的字符

（2）用户所输入的数字需符合程序要求

s 应当小于等于 n

k 应当小于 n

剩余人数 k 不能为 0 也不能等于 n

2.思路与设计

2.1 基本思路

此题用单向循环链表，链表中每一个元素表示一个人，用指针指向链表中的元素，循环遍历，当计数达到循环数时，将此时指针指向的人从链表中删除。计数清零，继续遍历，直到剩余人数符合要求。

2.2 设计

2.2.1 数据结构

单向循环链表。

2.2.2 结构体

```
struct man{
    int id;//序号
    man* next;//下一个人
};
```

新建一个结构体，表示一个人，存储数据包括代表此人的序号和指向后一个人的指针。

3.具体实现

3.1 输出离开旅客的序号

核心代码

```
//开始游戏
    for (cur=start; ;cur=cur->next) {
        static int count=1;//记录循环数
        static int dead=0;//记录死者个数
        if (count==m-1) {//如果下一个人将要死
            dead++;
            man* toDie=cur->next;
            if (toDie == data) {
                //如果是序号为1的人死去，要将data指针指向下一个人
                data = toDie->next;
            }
            cout<<"第"<<dead<<"个死者的位置是："<<toDie->id<<endl;
            cur->next = toDie->next;
            delete toDie;//释放空间
            count=1;//计数从1开始
            if (dead==n-k) {
                //幸存者数符合要求，结束游戏
                break;
            }
        } else {
            count++;//循环数+1
        }
    }
```

说明

用一个 cur 指针指向链表元素来模拟“数人”，起点是序号为 s 的人。for 循环执行一次，cur 指针就沿着链表的方向移动一个元素，计数 count 加一。
此循环链表是单向循环链表，但在链表中删除元素时，需要对该元素前后的元素进行修改，如果此时 cur 已经指向将要删除的元素，我们难以再找到它前面的元素，因此我们计数到（m-1）时就要暂停并进行删除元素的操作。则 count 加到（m-1）的时候，就要处理将被删除元素的前后元素的指针，然后在删除元素之前，先输出死者信息（即序号）。cur 指针继续后移，count 回到 1 开始计数。如果幸存者数已经符合要求，则退出 for 循环。游戏结束。

3.2 输出剩余旅客的序号

核心代码

```
cout<<"幸存者的位置为：
    //输出幸存者
    cur = data;
    for (int i=0; i<k; i++) {
        cout<<cur->id<<" ";
        cur = cur->next;
    }
```

说明

在前面已经保存了 data 指针，data 指针指向剩余旅客中序号最小的人。剩余旅客仍然保持一个循环链表结构，用 for 循环和 cur 指针可以对链表进行遍历，输出每一个剩余旅客的序号。

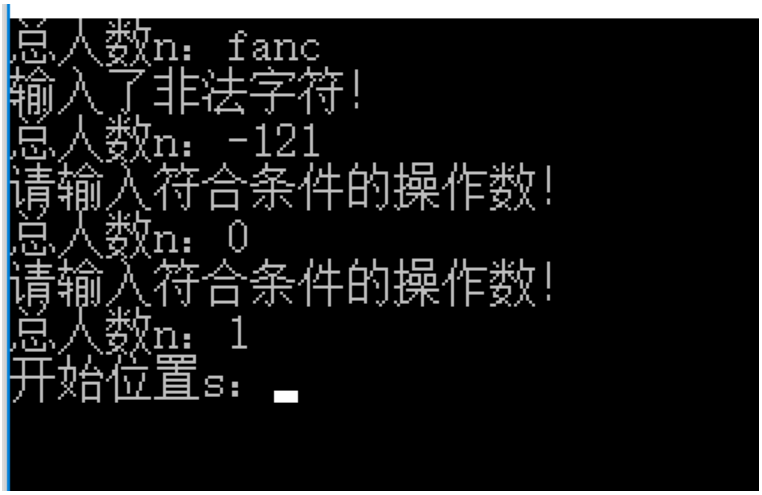
4.测试

4.1 功能测试

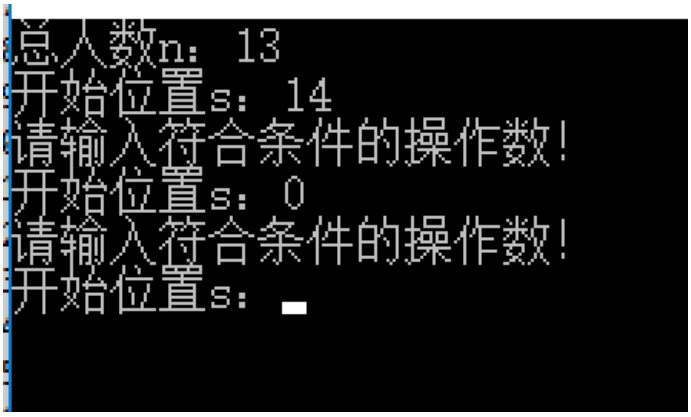
4.2 出错测试

4.2.1 输入非法字符

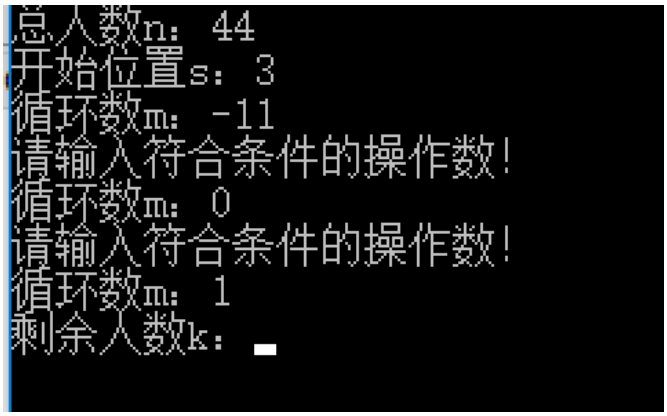
4.2.2 输入不符合条件的 n



4.2.3 输入不符合条件的 s



4.2.4 输入不符合条件的 m



4.2.5 输入不符合条件的 k