天津大学

数据结构上机实验报告

题目:用单循环链表解决约瑟夫问题

学生姓名 王雨朦

学生学号 2016229082

学院名称 国际工程师学院

专业 计算机

时间 2016/9/22

目　　录

第一章　需求分析

1.1 原题表述

一个旅行社要从 n 个旅客中选出一名旅客，为他提供免费的环球旅行服务。旅行社安排这些旅客围成一个圆圈，从帽子中取出一张纸条，用上面写的正整数 m(<n)作为报数值。游戏进行时，从第 s 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 的人被淘汰出列，然后从他顺时针方向上的下一个人开始重新报数，如此下去，直到圆圈中只剩下一个人，这个最后的幸存者就是游戏的胜利者，将得到免费旅行的奖励。其中数据结构采用**单循环链表**。

1.2 解决方案

1) 用单循环链表建立约瑟夫环；
2) 找到被指定的开始结点 s（链表中结点的查找）；
3) 从 s 结点开始,找到将要被淘汰的结点，删除并输出结果（链表结点的查找和删除）；
4) 找 n 次，结束。

第二章　概要设计

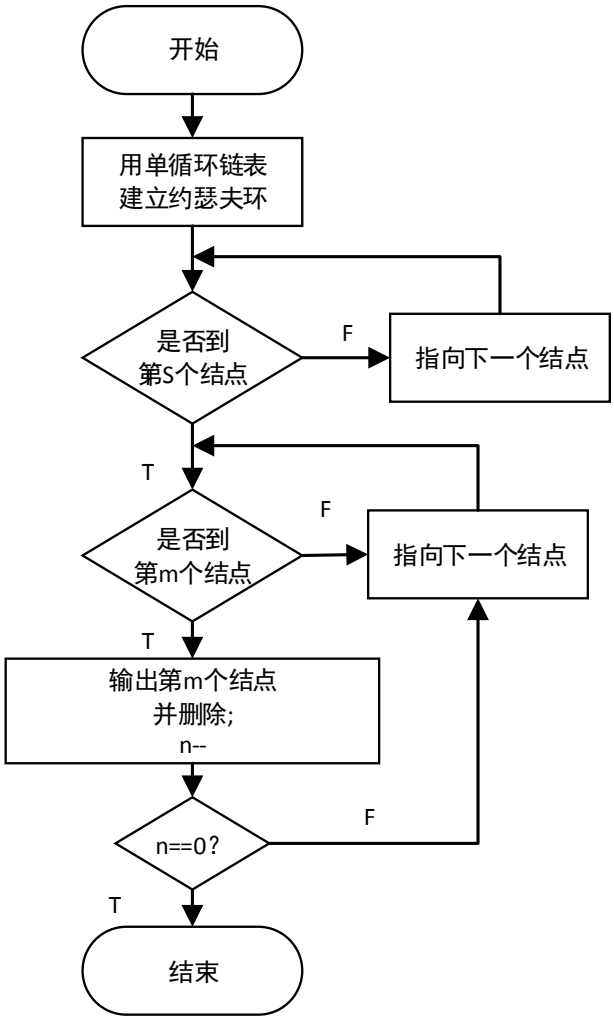
2.1 抽象数据类型

单循环链表:

```
typedef struct node{
    int data;
    struct node *next;
}node, *LinkList;

void Josephus(int n,int m, int s)
```

2.2 算法



第三章　详细设计

3.1 程序代码

```
#include<stdio.h>
#include<stdlib.h>

typedef struct node{
    int data;
    struct node *next;
}node, *LinkList;

void Josephus(int n,int m, int s){
    printf("******** Solve Josephus Problem ********\n");
    node *ha = (node *)malloc(sizeof(node));
    ha->data=1;
//initialize josephu-circle
    node *p = ha,*q;
    for(int i=2; i<=n; i++){
        q=(node *)malloc(sizeof(node));
        q->data=i;
        p->next=q;
        p=q;
    }
    p->next=ha;

//find start
    p=ha;
    for(int i=1;i<s;i++)
        p=p->next;

    while(n--){
        for(int i =1; i<m-1; i++)//the node before the delete node
            p=p->next;
        //delete q
        q=p->next;
        p->next =q->next;
        printf("Number %d is out.\n",q->data);
        free(q);
        p=p->next;
    }
```

```
        printf("***************END***************\n");
    }
    int main(){
        int n,m,s;
        //printf("Please enter n, m and s:\n");
        //scanf("%d%d%d", &n,&m,&s);
        printf("Enter n:\n");
        scanf("%d",&n);
        printf("Enter m:\n");
        scanf("%d",&m);
        printf("Enter s:\n");
        scanf("%d",&s);
    Josephus(n,m,s);
    return 0;
    }
```

第四章　调试分析

4.1 调试过程

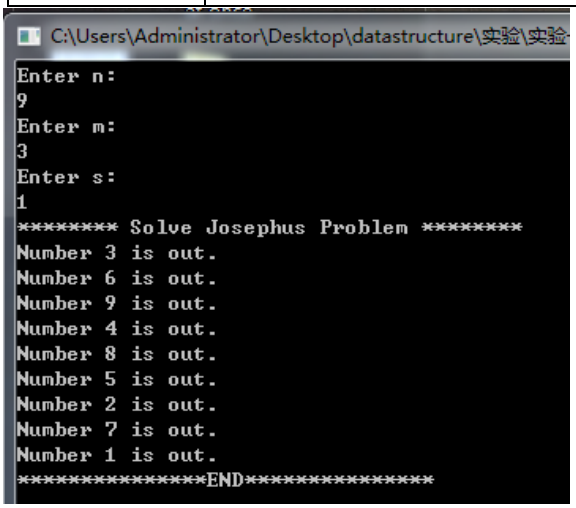
Bug 名称	结点删除逻辑错误
Bug 描述	当删除链表中第一个结点后，其它被删除的结点为第 m-1 个
Bug 原因	删除一个结点后，p 指针指向被删除的前一个结点，没有从下一个结点开始数 m 个
Bug 解决方案	当删除第 m 个结点后，p 指向下一个结点
Bug 总结	删除结点之后忽略了应从下一个开始

Bug 名称	输出格式错误
Bug 描述	忘记换行
Bug 原因	一句话结束没有输入换行符
Bug 解决方案	加上换行符\n
Bug 总结	习惯不好，应该养成好的习惯

第五章　测试结果

5.1 测试过程

测试编号	1
测试对象	n,m
测试输入参数	N=9,m=3
测试步骤	将 s 赋值为 1，观察结果
测试预期结果	3,6,9,2,4,8,5,2,7,1
测试输出结果	3,6,9,2,4,8,5,2,7,1
测试分析	程序正确



测试编号	2
测试对象	s
测试输入参数	S=2
测试步骤	N.,m 值与测试 1 相同，s 变为 2
测试预期结果	4 ,7,1,5,9,6,3,8,2
测试输出结果	4 ,7,1,5,9,6,3,8,2
测试分析	程序正确