实验报告

题目：用单循环链表编制模拟约瑟夫环的程序

班级：电信1809 姓名：刘俊逸 学号：U201813492 完成日期：2019.3.25

1. 需求分析
2. 本演示程序中，用单向循环链表储存结构模拟约瑟夫环过程，游戏者的数据储存在单项循环链表中，通过设计算法来对单向循环链表进行操作以达到模拟约瑟夫环过程。
3. 演示程序以用户和计算机的对话方式进行，即在计算机终端上显示“提示信息”之后，有用户在键盘上输入演示程序中规定的运算命令；相应的输入数据和运算结果显示在其后。
4. 程序执行的命令包括：
5. 构造结构体 2）构造单项循环链表 3）构造删除算法 4）构造开始算法 5）主函数
6. 测试数据

人数n=7，m的初值为6，7个人的密码依次为：3 1 7 2 4 8 4

1. 概要设计

为实现上述程序功能，应以单向循环链表表示约瑟夫环。

1. 详细设计
2. 构造结构体和记录删除数据的变量

typedef struct L

{

int num;//序号

int data;//密码

struct L \*next;

}L,\* LinkList;

int delate\_data = 0;//记录被删除结点的数据

1. 创建单向循环链表并存入玩家数据

int iniList()//初始化空链表

{

LinkList head;

if(head = (LinkList) malloc (sizeof(L))) head->next = NULL;

return head;

}

int creatList(int a[],int n)//创建单循环链表并存入数据

{

LinkList head = iniList();//创建一个空链表

LinkList r = head;

for(int i=0;i<n;i++)

{

LinkList q = (LinkList) malloc (sizeof(L));//创建新结点

q->num = i+1;//存入序号

q->data = a[i];//存入密码

r->next = q;//连接新创建的结点

r = q;

printf("序号:%d,密码为:%d\n",q->num,q->data);//显示存入的序号和密码

}

r->next = head->next;//将链表的末端连接到首端,形成单循环链表

return head;//返回存好数据的循环链表

}

1. 构造删除结点算法

int Delete(LinkList \*m,int i)//删除出列结点,m是需要删除的链表,i表示密码

{

LinkList p,q;

p = m;

for(int j=0;j<i-1;j++) p = p->next;//找到需要删除结点的前一个结点

q = p->next;//q指向需要被删除的结点

p->next = q->next;//删除结点

printf("序号:%d出局 \n",q->num);//显示每次出列玩家的序号

delate\_data = q->data;//记录删除结点的密码,下次使用

free(q);

return p;//导出已经进行了删除操作的链表

}

1. 构造游戏运行函数

int startGame(LinkList \*L,int n,int first\_m)

{

LinkList q = L;

int first = 1;

while(n-1)//剩下的人数不为1运行,直到只剩下一个人

{

if(first)//第一次开始

{

q = Delete(q,first\_m);

first = 0;//仅运行一次就不再运行

}

else q = Delete(q,delate\_data);

n--;//删除一次人数就少一个

}

return q->num;//导出最后一个人的序号

}

1. 构造主函数

void main()

{

int p[30],n=0,first\_m;

LinkList head;

printf("请依次输入初始m值:");

scanf("%d",&first\_m);

printf("请依次输入游戏者密码,输入非整数字符则表示结束\n");

while(scanf("%d",&p[n++]));//输入整数字符继续,输入非整数字符结束,并将密码存入数组p

printf("总参加人数为n:%d\n初始m值为:%d\n",n-1,first\_m);

head = creatList(p,n-1);

printf("游戏过程如下\n");

printf("最后一个出列者序号为:%d\n",startGame(head,n-1,first\_m));

}

1. 调试分析
2. 由于对单向循环链表的认识不足，在构造单项循环链表的早期版本中未设置成循环链表，仅构造了一个单链表。
3. 刚开始时曾忽略了一些变量参数的标识“&”，使调试程序时费时不少。今后应重视确定参数的变量和赋值属性的区分和标识。
4. 测试结果

初始m=6，密码为3 1 7 2 4 8 4，出列结果为6 1 4 7 2 3 5

初始m=5，密码为4 5 3 7 8 6 1 9 出列结果为5 6 4 8 1 2 3 7