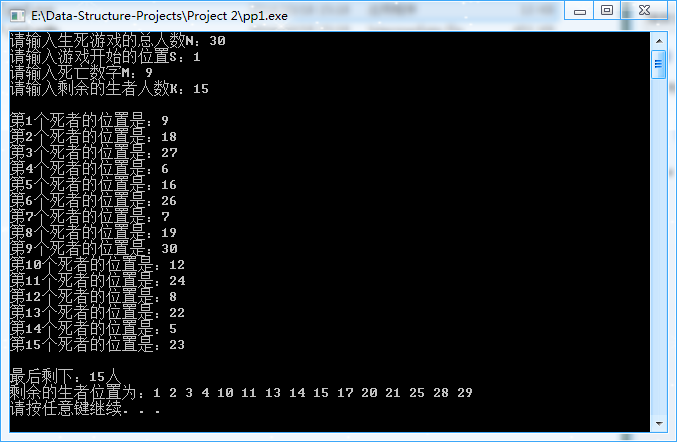
# 约瑟夫生者游戏

## 一.项目背景：

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。

## 二.项目功能

输入总人数N，开始位置S，死亡数字M，剩余人数K。程序将自动模拟游戏的进行过程，并输出结果。



## 三.程序描述

根据项目要求，由于有频繁的删除操作，故我选择了双向链表+循环链表的数据结构，类的声明如下：

class person

{

friend game;

private:

int num; //序号

person \*llink; //左儿子

person \*rlink; //右儿子

public:

person(int pos) :num(pos){}

};

然后建立链表：

for (int i = 2; i <= n; i++)

{

p = new person(i);

tail->rlink = p;

p->llink = tail;

p->rlink = NULL;

tail = p;

}

//将头指针与尾指针链接到一起

tail->rlink = head;

head->llink = tail;

在游戏过程中随时删除节点：

//如果删除的是头指针，则将头指针往后移一位

if (head->num == p->num)

head = head->rlink;

//删除当前的P指针

p->llink->rlink = p->rlink;

p->rlink->llink = p->llink;

temp = p;

p = p->llink;

delete temp;

以上就是该程序的思路及核心代码.