**约瑟夫环问题**

R6苏卓荣201330340621

**链表法：**

思路为，将所有人围成一个循环链表，然后根据要求不断抽取人，即抽取链表中的结点，直到抽到链表为空则停止

代码如下

#include <iostream>

#include <list>

#include <vector>

#define LI list<int>

#define VI vector<int>

using namespace std;

LI cycle;

VI order;

void next\_iter(LI::iterator &it)//利用STL中的双向链表模拟成循环链表，探测到iter等于链尾的时候，使iter指向链头

{

it++;

if(it == cycle.end()) it = cycle.begin();

}

void print\_order()

{

unsigned int i;

for(i=0;i!=order.size();i++)

cout<<order[i]<<' ';

cout<<endl;

}

void run()

{

int n,s,k;//n个人形成环，从第s个人开始，每隔k个人抽出一个人

int i;

cycle.clear();

order.clear();

cin>>n>>s>>k;

//建环

for(i=1;i<=n;i++)

cycle.push\_back(i);

LI::iterator it,it2;

//定位到第s个人

it = cycle.begin();

for(i=0;i<s-1;i++)

next\_iter(it);

//不断抽人，直到链表空

while(!cycle.empty())

{

for(i=0;i<k-1;i++)

next\_iter(it);//取下一个人的位置

order.push\_back(\*it);

it2=it;

next\_iter(it);

cycle.erase(it2);

}

print\_order();

}

int main()

{

run();

return 0;

}

**线性表法：**

思路为将每个人的编号按顺序存入一个线性表中，然后根据要求不断抽人，当指向最后一个人的时候，即数组最后一位的时候，则将指针指向回第一位，重复此过程直到线性表为空

代码如下（省略掉线性表的操作，只留下核心部分）

void next\_iter(SqList &L,int &it)

{

it++;

if(it>L.length) it=1;

}

void print\_order()

{

unsigned int i;

for(i=0;i!=order.size();i++)

printf("%d ",order[i]);

printf("\n");

}

void run()

{

int n,s,k,e,it;

int i;

SqList L;

order.clear();

scanf("%d%d%d",&n,&s,&k);

InitList\_Sq(L);

for(i=1;i<=n;i++) {ListInsert\_Sq(L,i,i);}

it=1;

for(i=0;i<s-1;i++)

next\_iter(L,it);

while(L.length)

{

for(i=0;i<k-1;i++)

next\_iter(L,it);

ListDelete\_Sq(L,it,e);

order.push\_back(e);

if(it>L.length) it=1;

}

print\_order();

}

int main()

{

run();

return 0;

}