假设除去第k个人。

0, 1, 2, 3, ..., k-2, k-1, k, ..., n-1　　//original sequence (1)

0, 1, 2, 3, ..., k-2,      , k, ..., n-1　　//get rid of kth person (2)

k, k+1, ..., n-1,    0,    1,        ..., k-2　　//rearrange the sequence (3)

0, 1,     ..., n-k-1, n-k, n-k+1, ..., n-2　　//the n-1 person (4)

我们假设f(n)的值为n个人中最后存活的人的序号，则

注意到(2)式(3)式(4)式其实是同一个序列。//这个很重要啊，要想清楚，这三个是同一个式子

注意(1)式和(4)式，是同一个问题，不同的仅仅是人数。

假设我们已知f(n-1)，即(4)式中最后剩下的人的序号，则(3)式所对应的序号，就是f(n)，即(1)式n个人中最后存活的序号。

而从(3)(4)式中我们不难发现有这样一个递推式：

f(n) = (f(n-1) + k) % n

显然，f(1) = 0。

于是递推得f(n)

因为是从m开始，所以递推的最后要单独列出来

普通的约瑟夫是从0开始

典型的约瑟夫问题，不过，这是从第m个人开始取得，而不是第1了，只需要最后+m即可然后，约瑟夫的递推公式是以0开始的，0-n-1而不是1-n，所以最后还要+1

#include<stdio.h>

#include<iostream>

using namespace std;

int main()

{

int n,k,m;

while(~scanf("%d %d %d",&n,&k,&m) && n)

{

int ans=0;

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

ans=(ans+k)%i;

ans=(ans+m)%n+1;

printf("%d\n",ans);

}

return 0;

}