# 16.题目描述 ：

有一个数组aNJ顺序存放0-N-1,要求每隔两个数删掉一个数,到末尾时循环至开头继续进行,求最后一个被删掉的数·的原始下标位置。以8个数(N=7)为例: {0, 1, 2, 3, 4, 5, 6, 7} , 0->1 >2(删除)->3->4->5(删除)->6->7->0(删除),如此循环直到最后一个数被删除。

## 输入描述:

每组数据为一行一个整数n(小于等于 1000),为数组成员数,如果大于 1000,则对a[999]进行计算。

## 输出描述:

一行输出最后一个被删掉的数的原始下标位置

## 示例：

输入：

8

输出：

6

import java.util.Scanner;

public class Main {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

while (scanner.hasNextInt()) {

int len = scanner.nextInt();

int[] index = new int[len];

System.out.println(countNum(len, index));

}

}

public static int countNum(int len, int[] index) {

int i = 0;

int count = 0;

int count1 = 0;

while (true) {

if (count1 == len) {

return i - 1;

}

if (i == len) {

i = i % len;

}

if (index[i] != -1) {

count++;

}

if (count == 3) {

count = 0;

count1++;

index[i % len] = -1;

}

i++;

}

}

}