



- [ACTIVITIES](#)
- [OLYMPIAD](#)
  - [EVENTS](#)
  - [MEMBERS](#)
  - [ARCHIVES](#)
  - [ABOUT](#)

**Related:**

- [Overview](#)
- [Current](#)
- [Archives](#)
- 

## Indian National Olympiad in Informatics, 2003

[IARCS home](#) > [OLYMPIAD](#) > [Archives](#)

### INOI-2003, Sample questions

#### Problem 1: You can't get there from here

In a primitive video game, a spot bounces around within a rectangular grid. The southwest corner of the grid has coordinates  $(0,0)$  and the northeast corner has coordinates  $(r,c)$  where  $0 < r \leq 50$  and  $0 < c \leq 50$ . The southeast corner has coordinates  $(0,c)$  and the northwest corner has coordinates  $(r,0)$ . The spot always travels on the diagonal; that is, in one of the directions NE, NW, SE, SW. The outer edges of the grid serve as mirrors: after visiting a position on the edge of the grid the spot "bounces" off according to the normal rules of reflection (Snell's Law). For example, if the spot were travelling NE and hit the east edge of the grid, it would change direction to NW. If the spot were to hit the corner of the grid it would change to the opposite direction.

Given a grid size, two points  $A$  and  $B$  lying on the grid, and an initial direction, you are to determine if the spot moves from  $A$  to  $B$  and, if so, how far the spot moves (in terms of number of grid positions) before reaching  $B$  the first time.

#### Input Specification

The input consists of an integer  $n$ , followed by  $n$  data sets. Each data set begins with a line containing  $r$  and  $c$ , followed by two lines containing the coordinates of points  $A$  and  $B$  respectively, followed by one line containing NE, NW, SE, or SW - the initial direction of travel.

### Output Specification

For each case, print a sentence as shown below indicating whether or not  $B$  can be reached, and, if it can, how far the spot moves before reaching  $B$ .

### Sample Input

```
2
3 4
0 0
0 4
NE
4 2
3 1
3 2
NW
```

### Output for Sample Input

B can be reached from A after 12 move(s). B cannot be reached from A.

Copyright (c) IARCS 2003; Last Updated: 11 Apr, 2003



- [ACTIVITIES](#)
- [OLYMPIAD](#)
  - [EVENTS](#)
  - [MEMBERS](#)
  - [ARCHIVES](#)
  - [ABOUT](#)

**Related:**

- [Overview](#)
- [Current](#)
- [Archives](#)
- 

## Indian National Olympiad in Informatics, 2003

[IARCS home](#) > [OLYMPIAD](#) > [Archives](#)

### INOI-2003, Sample questions

#### Problem 2: Digital rivers

A digital river is a sequence of numbers where the number following  $n$  is  $n$  plus the sum of its digits. For example, 12345 is followed by 12360 since  $1+2+3+4+5 = 15$ . If the first number of a digital river is  $k$ , we call it river  $k$ .

For example, river 480 is the sequence beginning  $\{480,492,507,519,\dots\}$  and river 483 is the sequence beginning  $\{483,498,519,\dots\}$ .

Normal streams and rivers can meet, and the same is true for digital rivers. This happens when two digital rivers share some of the same values. For example, river 480 meets river 493 at 519, meets river 507 at 507, and never meets river 481.

It is known that every digital river will eventually meet river 1, river 3 or river 9. Write a program that reads an integer  $n$  ( $1 \leq n \leq 16384$ ) and prints out the value where river  $n$  first meets one of these three rivers, as well as which of the three rivers it meets at this value (1, 3 or 9).

## Sample Input

86

## Sample Output

101 1

Copyright (c) IARCS 2003; Last Updated: 11 Apr, 2003



- [ACTIVITIES](#)
- [OLYMPIAD](#)
  - [EVENTS](#)
  - [MEMBERS](#)
  - [ARCHIVES](#)
  - [ABOUT](#)

**Related:**

- [Overview](#)
- [Current](#)
- [Archives](#)
- 

## Indian National Olympiad in Informatics, 2003

[IARCS home](#) > [OLYMPIAD](#) > [Archives](#)

### INOI-2003, Sample questions

#### Problem 3: Recurring decimals

Write a program that reads an integer  $n$  as input and prints out the decimal representation of  $1/n$ . If  $1/n$  is a recurring decimal, print out *two* copies of the recurring portion. You may assume that  $n$  is between 1 and 10000.

Here are some sample inputs and outputs for this problem:

- *Input: 2*  
*Output: 0.500*
- *Input: 3*  
*Output: 0.33*
- *Input: 5*  
*Output: 0.200*
- *Input: 7*  
*Output: 0.142857142857*
- *Input: 11*

*Output: 0.0909*

Copyright (c) IARCS 2003; Last Updated: 11 Apr, 2003