

Indian National Olympiad in Informatics, 2014

Time: 3 hours

19 January, 2014

Instructions

- (a) You will have to return this question paper at the end of the examination with relevant parts filled out.
- (b) There are two questions. You have to write working programs in Pascal, C or C++ to solve each of these questions.

- *Only your source code will be submitted for evaluation. Your program will be re-compiled and run on the evaluation computer.*
- *Make sure your C/C++ programs compile with the GNU C compiler (`dev-cpp` or `Code::Blocks` under Windows). Programs written using Turbo C++ may not compile and run in the evaluation environment and may hence lose marks.*

If you work in C/C++, make sure you do not write `#include <conio.h>` in your program or use any functions defined in `conio.h`. If you do so, your program will not compile and you will get no marks.

- (c) At the end of each question, there is a space to indicate the location of the source code file for your solution. Please fill up this information without fail. Otherwise, your solution cannot be evaluated.
- (d) All input for your programs will come from the keyboard. All output from your programs should be written to the screen.
- (e) *On the computer where you are working, you will find a collection of sample inputs and outputs for each problem to test your programs. Please ask your centre supervisor if you cannot locate these sample inputs and outputs.*

This set of sample inputs is not exhaustive. Additional test cases will be used for final evaluation. Even if your program correctly solves all these sample inputs, there is no guarantee that your program will pass all the cases in the final evaluation.

- (f) Please fill out your contact details on the reverse of this page as completely as you can. Use your serial number as reported in the ZIO/ZCO results page as your roll number for INOI. If you qualified under multiple categories, use the ZIO serial number as your roll number. Ask your centre supervisor if you do not know your roll number.

...P.T.O.

- (g) At the beginning of each program, include a comment with your name, roll number and the problem number. For instance, if your name is Sachin Tendulkar, your roll number is 107 and the problem you are writing a C/C++/Pascal program for is Problem 1, add a comment of the form

```
/* Sachin Tendulkar, 107, Problem 1 */
```

at the top of your file.

This will help us if files accidentally get mixed up when being sent from the exam centres to the evaluation centre.

Contact details	
Roll Number:	
Name:	
Home Address:	
	PIN Code
Phone Number(s): (with STD code)	
Email address:	

Question 1 Highway bypass

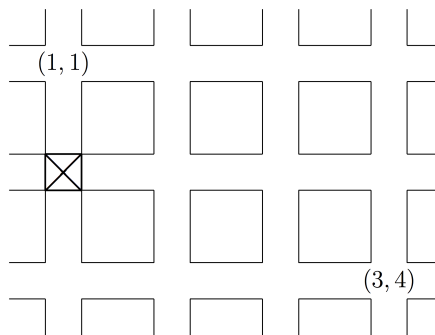
Due to resurfacing work, all north-south traffic on the highway is being diverted through the town of Siruseri. Siruseri is a modern, planned town and the section of roads used for the diversion forms a rectangular grid where all cars enter at the top-left intersection (north-west) and leave at the bottom-right intersection (south-east). All roads within the grid are one-way, allowing traffic to move north to south (up-down) and west to east (left-right) only.

The town authorities are concerned about highway drivers overspeeding through the town. To slow them down, they have made a rule that no car may travel more than d consecutive road segments in the same direction without turning. (Closed-circuit TV cameras have been installed to enforce this rule.)

Of course, there is also repair work going on within the town, so some intersections are blocked and cars cannot pass through these.

You are given the layout of the rectangular grid of roads within Siruseri and the constraint on how many consecutive road segments you may travel in the same direction. Your task is to compute the total number of paths from the entry (top-left) to the exit (bottom-right).

For instance, suppose there are 3 rows and 4 columns of intersections, numbered from (1,1) at the top-left to (3,4) at the bottom-right, as shown on the right. Intersection (2,1) in the second row, first column is blocked, and no car may travel more than 2 consecutive road segments in the same direction.



Here, $(1,1) \rightarrow (1,2) \rightarrow (2,2) \rightarrow (3,2) \rightarrow (3,3) \rightarrow (3,4)$ is a valid path from (1,1) to (3,4), but $(1,1) \rightarrow (1,2) \rightarrow (1,3) \rightarrow (1,4) \rightarrow (2,4) \rightarrow (3,4)$ is not, because this involves 3 consecutive road segments from left to right. The path $(1,1) \rightarrow (2,1) \rightarrow (2,2) \rightarrow (2,3) \rightarrow (3,3) \rightarrow (3,4)$ is ruled out because it goes through a blocked intersection. In this example, you can check that the total number of valid paths is 5.

Input format

- Line 1 : Three space-separated integers, R , C and d , where R is the number of rows in the grid, C is the number of columns in the grid and d is the maximum number of consecutive segments you can travel in any direction.
- Lines 2 to $R+1$: Each line contains C integers, each of which is 0 or 1, describing one row of intersections in the grid. An intersection marked 0 is blocked and an intersection marked 1 is available to pass through. The start (top-left) and finish (bottom-right) intersections are always marked 1.

Output format

A single integer—the number of paths from the top-left intersection to the bottom-right intersection that go only down and right, and obey the d constraint.

Since the final answer may not fit in a variable of type `int`, report your answer modulo 20011. *Be careful to avoid overflows in intermediate computations.*

Test Data

The testdata is grouped into three subtasks. In all subtasks, $1 \leq R \leq 300$, $1 \leq C \leq 300$ and $1 \leq d \leq 300$. In addition, each subtask has the following constraints on the inputs.

- Subtask 1 (20 marks) : $d = \max(R, C) - 1$. (In other words, there is no limit on the number of consecutive segments you can travel in one direction.)
- Subtask 2 (30 marks) : $d = 2$.
- Subtask 3 (50 marks) : No additional constraint on d .

Example

Here is the sample input and output corresponding to the example above.

Sample input

```
3 4 2
1 1 1 1
0 1 1 1
1 1 1 1
```

Sample output

```
5
```

Time and memory limits

The time limit for this task is 3 seconds. The memory limit is 512 MB.

Note: Your program should not print anything other than what is specified in the output format. Please remove all diagnostic print statements before making your final submission. A program with extraneous output will be treated as incorrect!

Important

Indicate the location of the source code file for your solution to Question 1 in the box below. *If you fail to do this, your solution cannot be evaluated!*

Source file:

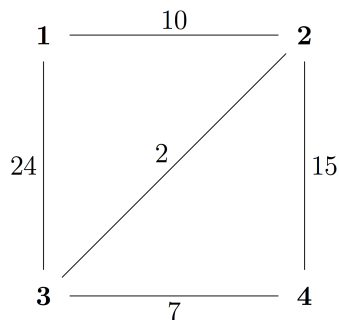
Question 2 Free ticket

Nikhil's slogan has won the contest conducted by Drongo Airlines and he is entitled to a free ticket between any two destinations served by the airline. All cities served by Drongo Airlines can be reached from each other by some sequence of connecting flights. Nikhil is allowed to take as many connecting flights as needed, but he must take the cheapest route between his chosen destinations.

Each direct flight between two cities has a fixed price. All pairs of cities connected by direct flights have flights in both directions and the price is the same in either direction. The price for a sequence of connecting flights is the sum of the prices of the direct flights along the route.

Nikhil has information about the cost of each direct flight. He would like to maximize the value of his prize, so he would like to choose a pair of cities on the network for which the cost of the cheapest route is as high as possible.

For instance, suppose the network consists of four cities $\{1, 2, 3, 4\}$, connected as shown on the right.



In this case, Nikhil should choose to travel between 1 and 4, where the cheapest route has cost 19. You can check that for all other pairs of cities, the cheapest route has a smaller cost. For instance, notice that though the direct flight from 1 to 3 costs 24, there is a cheaper route of cost 12 from 1 to 2 to 3.

Input format

- Line 1 : Two space-separated integers, C and F . C is the number of cities on the network, numbered $1, 2, \dots, C$. F is the number of pairs of cities connected by a direct flight.
- Lines 2 to $F + 1$: Each line describes one direct flight between a pair of cities and consists of three integers, x , y and p , where x and y are the two cities connected by this flight and p is the price of this flight.

For all direct flights, $x \neq y$ and no pair of cities is connected by more than one direct flight. If there is a direct flight from x to y with price p , there is also a flight from y to x with price p and exactly one of these two will be listed.

Output format

The output consists of a single integer, the maximum cost among the cheapest routes between all pairs of cities across the airline's network.

Test Data

The testdata is grouped into three subtasks. In all subtasks, $2 \leq C \leq 230$ and $1 \leq F \leq 26335$. In addition, each subtask has the following constraints on the inputs.

- Subtask 1 (20 marks) : $F = C - 1$ (that is, the airline network is a tree) and $p = 1$ for each direct flight.
- Subtask 2 (30 marks) : There is no constraint on the shape of the network, but for each direct flight, $p = 1$.
- Subtask 3 (50 marks) : There is no constraint on the shape of the network, but for each direct flight, $0 \leq p \leq 10^5$.

Example

Here is the sample input and output corresponding to the example above.

Sample input

```
4 5
1 2 10
1 3 24
2 3 2
2 4 15
3 4 7
```

Sample output

```
19
```

Time and memory limits

The time limit for this task is 3 seconds. The memory limit is 256 MB.

Note: Your program should not print anything other than what is specified in the output format. Please remove all diagnostic print statements before making your final submission. A program with extraneous output will be treated as incorrect!

Important

Indicate the location of the source code file for your solution to Question 2 in the box below. *If you fail to do this, your solution cannot be evaluated!*

Source file: