ACM South American Regional

Collegiate Programming Contest

Warm-up Session

November 14, 1998

São Paulo – Brazil
Antofagasta – Chile
Caracas – Venezuela

PROBLEM 1: Generators

File Names

Source File: gen.pas or gen.c or gen.cpp
Input File: gen.in
Output File: gen.out

Statement of the Problem

Given a prime number p, we define $Z_p = \{0, 1, \ldots, p-1\}$. In this set we have the operator $\otimes$ defined as $x \otimes y = (x * y)\ mod\ p$.

We know that $(Z_p - \{0\}, \otimes)$ is a cyclic group, that is, there exists at least one element $x \in Z_p - \{0\}$ (called a generator of the group) such that for every $y \in Z_p - \{0\}$ there is an integer i such that $x \otimes x \otimes \ldots \otimes x$ (i times) is equal to y. For example, we know that 2 is a generator in $Z_5 = \{1, 2, 3, 4\}$, since $2 \otimes 2 = 4$, $2 \otimes 2 \otimes 2 = 3$, $2 \otimes 2 \otimes 2 \otimes 2 = 1$ and $2 \otimes 2 \otimes 2 \otimes 2 \otimes 2 = 2$.

In this problem you have to write a program to determine a generator of a group $(Z_p - \{0\}, \otimes)$, for a given prime number p.

Input Format

The input file may contain several instances of the problem. Each instance consists of a single prime number in one line. You may assume that each the number is not larger than 100.

The input file is terminated by a line containing the integer zero.

Output Format

For each instance of the problem, you program should print one line containing the prime number p for that instance, followed by any generator for $(Z_p - \{0\}, \otimes)$. The numbers may be separated by an arbitrary number of blank spaces.

The output of each instance is terminated by a blank line.

Sample Input

```
7
5
11
0
```

Sample Output

```
7   3

5   2

11  2
```

PROBLEM 2: Geometric Kingdoms

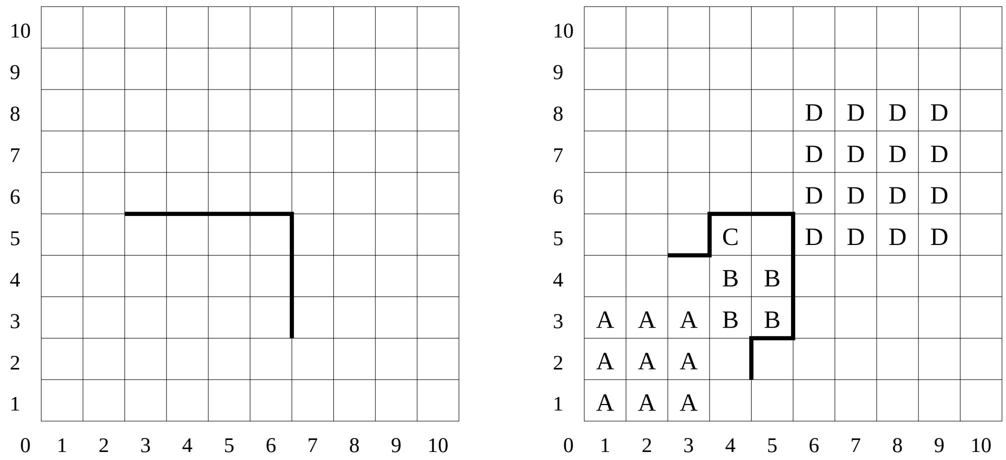
File Names

Source File: kingdom.pas or kingdom.c or kingdom.cpp
Input File: kingdom.in
Output File: kingdom.out

Statement of the Problem

In a far away land called Geometric, the inhabitants love the geometric perfection of straight lines and squares. Over the years, they built a network of roads that cross the land horizontal and vertically. These roads are uniformly distributed at every kilometer, and the citizens of Geometric used them to go from one point to another.

According with an ancient tradition, when traversing their mesh, if the citizens can choose between travelling horizontally or vertically, they always prefer to go horizontally. Figure 1(a) shows one such traversal, between point (2,6) and point (6,3).



(a) (b)

Figure 1: Paths in the kingdom

Recently, however, some power hungry people have started building fortresses and proclaiming themselves kings over certain areas. Being geometrics, all these area are squares Figure 1(b) shows four kingdoms: kingdom A is located at (0,0) and has side 3, kingdom B is at (3,2) and has side 2, kingdom C is at (3,4) with side 1 and kingdom D is at (5,5) with side 4. As these kingdoms do not intersect each other, until now there has been no war between them.

The peaceful citizens of Geometric do not want war at all; so, they decided to avoid using the roads that are inside a kingdom. They always prefer to use lines that do not belong to any kingdom. They may use borderlines though, but only when these lines are not part of a frontier between two neighboring kingdoms. But they are also lazy people, so they want always to use the shortest path between two points. In Figure 1(b) such a path is shown between points (2,4) and (4,1). Note that the borderlines of the land Geometric can also be used.

You should write a program that helps the citizens of Geometric to find the shortest path between two given points, while avoiding the belligerent kingdoms.

Input Format

The input file may contain several instances of the problem. Each instance has the following format (all numbers are positive integers):

1. The first line contains two integer numbers n and k: n corresponds to the dimension of Geometric (i.e. the number of rows and columns), and k is the number of kingdoms. You may assume that $n \leq 30$ and $k \leq 20$.
2. The following k lines contain each the description of one kingdom in the form of three integers x y d, where (x, y) is the coordinate of the lower left corner of the kingdom and d is its dimension.
3. The last line contains four integers x_0 y_0 x_1 y_1 corresponding to the coordinates of the starting point of the traversal, (x_0, y_0), and the ending point, (x_1, y_1).

You can assume that: (i) the kingdoms do not intersect, except for borderlines; (ii) the starting and ending points are not the same and are never located inside a kingdom.

The input file is terminated by a line containing the integers `0 0` (two zeroes).

All numbers are separated by an arbitrary number of blank spaces.

Output Format

For each instance of the problem, your program should print either the message "No path", if no path exists, or the coordinates of the path, one per line, beginning with the starting point and then listing all **turning points** until the ending point. If you found more than one path, you only need to list one of them, at your will.

Each instance is terminated with a blank line.

Sample Input

The program shown in Figure 1 has the following input:

```
10 4
0 0 3
3 2 2
3 4 1
5 5 4
2 4 4 1
0 0
```

Sample Output

The corresponding output is

```
2 4
3 4
3 5
5 5
5 2
4 2
4 1
```