BLACK BOX (Solution)

Authors: Cesar Cepeda (México), Hugo Ryckeboer (Argentina)

PROGRAMATIC SOLUTION: Black box is an output-only task, this means that the student does not have to submit a program as a solution, nevertheless this document briefly describes how the task could be solved by using a program.

The first step is to locate the leftmost deflector of the first row, this can easily be done as follows

- *Throw in left 1*. This can lead us to one the following three cases
  1. *The ball appears with no hits*: This means that there are no deflectors in this row.
  2. *The ball appears after one hit*: The value in *holeOut* gives you the column of the first deflector, its position can be deduced from the side where the ball appeared.
  3. *The ball appears after more than one hit*: It means that the ball was sent down and appeared after many collisions, the side and hole where it appears give no meaningful information. In this case it means that the leftmost deflector of the first row is in the position '\', so toggling it will cause the ball to go up. Execute the following operations **`Reset, Invert, Throw in left 1`**. The ball must appear after one hit in the top side and *holeOut* gives you the column of the leftmost deflector of the first row.

After the first deflector you must locate additional deflectors in the first row. This must be done proceeding in a recursive way.

Up to this point your partial knowledge of the box can be described as `....\?????????`, to explore farther to the right you should be able to send a ball that enters the first unknown square from the left.

- If the original position of the rightmost deflector known in the row is `\`, sending the ball is easy, you need reset the box and throw a ball from the top side through the hole where that deflector is. The deflector will change the direction of the ball and you can proceed as you did to find the first deflector.
- If the original rightmost deflector known in the row is `/`, in order to send a ball that enters the first unknown square from the left you have to toggle this deflector first. Since you know all the original positions of the deflectors to the left of that you can govern their position in a recursively way. Suppose that you have the following state `.../../../?????` the deflectors are in columns 4, 7, and 10. You have to toggle the deflector in column 10 in order to explore the unknown squares, this can be done by toggling all deflectors or by hitting the deflector in column 10 from the left. Toggling all deflectors requires that the program has a way of memorizing the toggles, and makes it more difficult to code. On the other hand you can toggle all deflectors with the following sequence: **`throw in left 1, throw in top 4, throw in top 7,`** after the last throw the deflector in column 10 is in the correct position to continue exploring.

Proceeding this way you can complete the first row.

How to analyze the second row and further.

In order to be able to analyze beyond the first row you must be able to put all the deflectors in the already known rows so that any ball that reaches the already known areas never bounces back into unknown parts of the box.

Suppose that you have already discovered all the deflectors in the first row and that the map of the box is as follows

```
.\\.././/\..
???????????
```

In order to find the leftmost deflector in the second row you should position all the deflectors in the first row as follows `.//.././/..` in that way, every ball that goes up from the second row to the first will eventually exit from the top or right side, and you will be able to know the column where the cross from second to first row was made.

The way in which you position the deflectors depends on what parts of the box you know, for example, lets say that you have discovered the following

```
.\\.././/\..
..\????????
```

now, the best way to position the deflectors in the first row is the following, every deflector to the left of the first unknown column in the second row shall be in the position `\` and every deflector on top and to the right of the first unknown column of the second row shall be in the position `/`.

Arranging deflectors as described you can explore the second row in a similar way to that of the first row.

The effort that it takes to explore each new row grows exponentially, because in order to explore new columns in each row you have to arrange all the deflectors in the previous positions. The best way to explore a box is from its four sides as described, that is, first row, first column, last row, last column. Then proceed to the second row, second column, etc. That way if you have *n* unknown elements in the first row you will have only *n-2* unknown elements in the second.

Care should be taken when exploring on different sides, because the meaning of `/` and `\` reverses depending on the side.