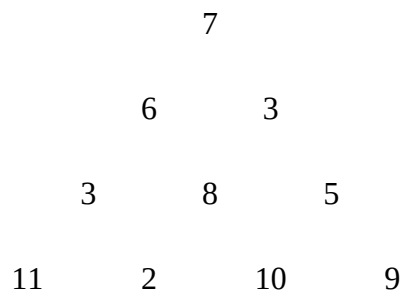
Roboreus Java Developer Programming Test #2.0.4:

Minimum Triangle Paths

Background

Consider the following triangle of numbers:



A **path** through the triangle is a sequence of adjacent nodes, one from each row, starting from the top. So, for instance, 7 → 6 → 3 → 11 is a path down the left hand edge of the triangle.

A **minimal path** is one where the sum of the values in its nodes is no greater than for any other path through the triangle. In this case, 7 + 6 + 3 + 2 = 18 is a minimal path.

We can store the triangle in a text file with each row on a separate line, and spaces between the numbers. Thus the triangle above would be stored in **text format** as:

```
7
6 3
3 8 5
11 2 10 9
```

Task

Write a **command-line** program in **Java** that reads a **text-format triangle** from **standard input** and outputs a **minimal path** to **standard output** as follows:

```
$ cat << EOF | java MinTrianglePath
> 7
> 6 3
> 3 8 5
> 11 2 10 9
> EOF
Minimal path is: 7 + 6 + 3 + 2 = 18
```

Besides producing correct answers, your code should:

- have good error handling – including being able to tell the end user whether and exactly where in the input there is an error (eg line too short or too long or invalid value)
- be clear, easy to follow and maintainable – ie it should be "production quality"
- be capable of producing the correct answer for a 500-row triangle in less than 0.5 seconds on a normal PC. (We will compile and run your code in a Cygwin/Windows environment.)

Notes

There is no time limit. As a guide, we wouldn't envisage the task taking longer than two to four hours. However, we would prefer a good solution to a rushed one.