

## Roboreus Java Developer Programming Test #2.0.6:

### Minimum Triangle Paths

#### **Background**

Consider the following triangle of non-negative integers:

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

A **path** through the triangle is a sequence of adjacent nodes, one from each row, starting from the top. So, for instance,  $7 \rightarrow 6 \rightarrow 3 \rightarrow 11$  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 (using a file containing the triangle above):

```
$ java MinTrianglePath < testfile.txt
Minimal path is: 7 + 6 + 3 + 2 = 18
```

An average PC should be able to produce the answer for a 500-line triangle within 0.5 seconds. Be aware that your code will be reviewed by a human and so your source files should be clear, easy to follow and maintainable, as well as your solution generating correct answers within the time bound.

We will be looking for the ability to handle edge and error cases – such as 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).