Connect in Scala

Readme (readme)	Test Suite (../connect)

Connect

Compute the result for a game of Hex / Polygon

The abstract boardgame known as
Hex (https://en.wikipedia.org/wiki/Hex_%28board_game%29) / Polygon /
CON-TAC-TIX is quite simple in rules, though complex in practice. Two players
place stones on a rhombus with hexagonal fields. The player to connect his/her
stones to the opposite side first wins. The four sides of the rhombus are
divided between the two players (i.e. one player gets assigned a side and the
side directly opposite it and the other player gets assigned the two other
sides).

Your goal is to build a program that given a simple representation of a board
computes the winner (or lack thereof).

The boards look like this (with spaces added for readability, which won't be in
the representation passed to your code):

```
1 . O . X .
2  . X X O .
3   O O O X .
4    . X O X O
5     X O O O X
```

White (O) plays from top to bottom, black (X) plays from left to right. In
the above example white has made a connection from left to right but nobody has
won since white didn't connect top and bottom.

The Scala exercises assume an SBT project scheme. The exercise solution source
should be placed within the exercise directory/src/main/scala. The exercise
unit tests can be found within the exercise directory/src/test/scala.

To run the tests simply run the command `sbt test` in the exercise directory.

For more detailed info about the Scala track see the help
page (http://help.exercism.io/getting-started-with-scala.html).

Source

view source

exercism.io
(/)

Beta

About (/about) - Donate (/donate)

 GitHub (https://github.com/exercism/exercism.io)   Twitter (https://twitter.com/exercism_io)

 Newsletter (https://tinyletter.com/exercism)

SPONSORS

 (https://bugsnag.com/blog/bugsnag-loves-open-source)

 (http://www.rackspace.com/)   (http://www.shopify.com/)

© 2015 Katrina Owen