

CSCE 314 HW 3 comment

Quan, Robert

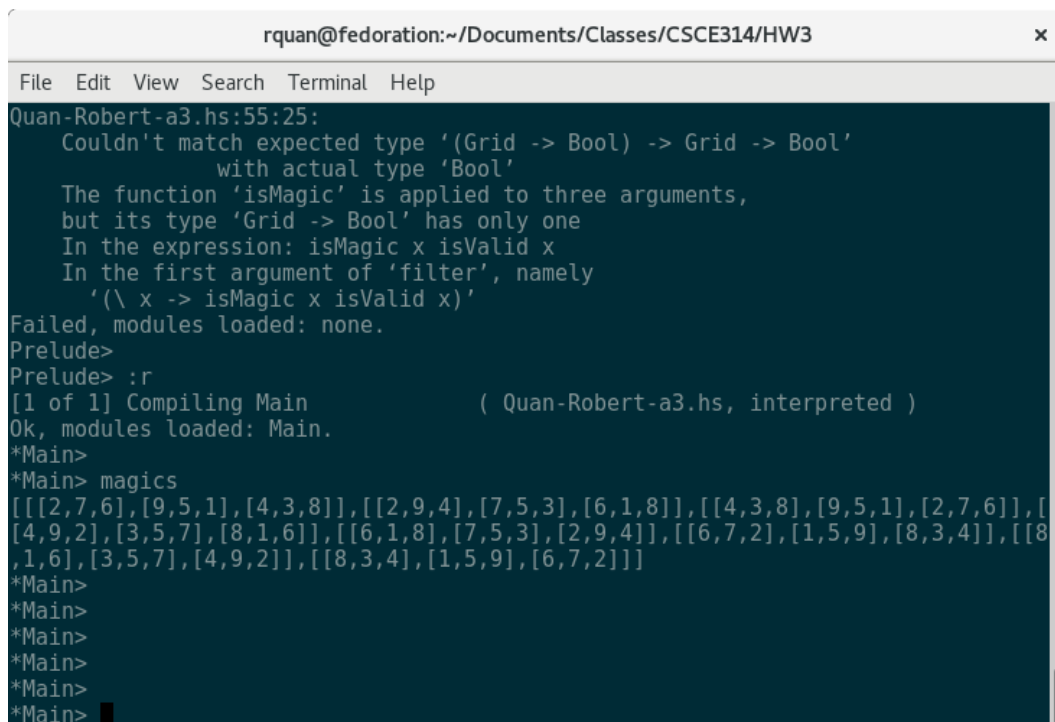
Department of Physics and Astronomy, Texas A&M University, College Station, Texas 77843-4242, USA

(Dated: June 27, 2017)

Submit a pdf file named hw3comment.pdf. In that file, describe how you can make your Sudoku Puzzle solver more interactive and more efficient. Describe your plan for implementation (you can mention the Haskell packages for doing that).

I. ANSWER I

To make the Sudoku Puzzle solver more efficient, it would be beneficial to set the initial conditions on the choices function to have even numbers only on the corners of the magic square and the number 5 in the middle. From the research I performed online, all magic squares have these two conditions. This greatly reduces the number of computations needed to find the squares from 9^9 to 4^8 , which is 3 orders of magnitude more efficient to compute. If we were to implement parallel computing, then we can assign different initial conditions to each processor to improve the time spent on computing the outputs. On my laptop, only one of the eight cores are used to compute magics, which took me around 35 minutes to obtain all eight squares.



```
quan@fedoration:~/Documents/Classes/CSCE314/HW3
File Edit View Search Terminal Help
Quan-Robert-a3.hs:55:25:
  Couldn't match expected type '(Grid -> Bool) -> Grid -> Bool'
    with actual type 'Bool'
  The function 'isMagic' is applied to three arguments,
  but its type 'Grid -> Bool' has only one
  In the expression: isMagic x isValid x
  In the first argument of 'filter', namely
    '(\ x -> isMagic x isValid x)'
Failed, modules loaded: none.
Prelude>
Prelude> :r
[1 of 1] Compiling Main          ( Quan-Robert-a3.hs, interpreted )
Ok, modules loaded: Main.
*Main>
*Main> magics
[[[2,7,6],[9,5,1],[4,3,8]],[[2,9,4],[7,5,3],[6,1,8]],[[4,3,8],[9,5,1],[2,7,6]],[
[4,9,2],[3,5,7],[8,1,6]],[[6,1,8],[7,5,3],[2,9,4]],[[6,7,2],[1,5,9],[8,3,4]],[[8
,1,6],[3,5,7],[4,9,2]],[[8,3,4],[1,5,9],[6,7,2]]]
*Main>
*Main>
*Main>
*Main>
*Main>
*Main>
```