Programming Assignment #1 - Cracker Barrel Game Solver
Patrick Sheehan
CSCE 420
22 January 2014
Sources: haskell.org, stackoverflow.com, piazza,
http://www.danobrien.ws/PegBoard.html - referenced the correct answers, not the code

This program was written in Haskell. Generally, a board is evaluated to see if there are any possible moves that can be made. If there aren't any, it is checked to see if it is a success (one peg anywhere, and one peg in a corner). If there are possible moves, evaluate all of them in this recursive pattern. A board is represented as a list of 15 pegs. a peg is a 4-tuple of integers: (row, column, peg number, and if it's full/empty/invalid). The return type of the recursive functions are 'Wins' which is a tuple of representing the total number of successes of a board from every subsequent board (if any), including itself.

The source file is: pa1-v3.hs

There are two ways to run it:

1.) compile: ghc pa1-v3.hs
and run: ./pa1-v3

```
patricks-mbp-4:PA1 v3 PatrickSheehan$ ghc pa1-v3.hs
patricks-mbp-4:PA1 v3 PatrickSheehan$ ./pa1-v3
(29760,6816)
```

2.) open the interpreter: ghci pa1-v3.hs
run main at the prompt: main

```
dhcp-10-203-48-144:PA1 v3 PatrickSheehan$ ghc pa1-v3.hs

[1 of 1] Compiling Main             ( pa1-v3.hs, pa1-v3.o )
Linking pa1-v3 ...
dhcp-10-203-48-144:PA1 v3 PatrickSheehan$ ghci pa1-v3.hs
GHCi, version 7.6.3: http://www.haskell.org/ghc/  :? for help
Loading package ghc-prim ... linking ... done.
Loading package integer-gmp ... linking ... done.
Loading package base ... linking ... done.
Ok, modules loaded: Main.
[1 of 1] Compiling Main             ( pa1-v3.hs, interpreted )
```

```
Ok, modules loaded: Main.

*Main> main
(29760,6816)


An Aggie does not lie, cheat, steal, or tolerate those who do
```