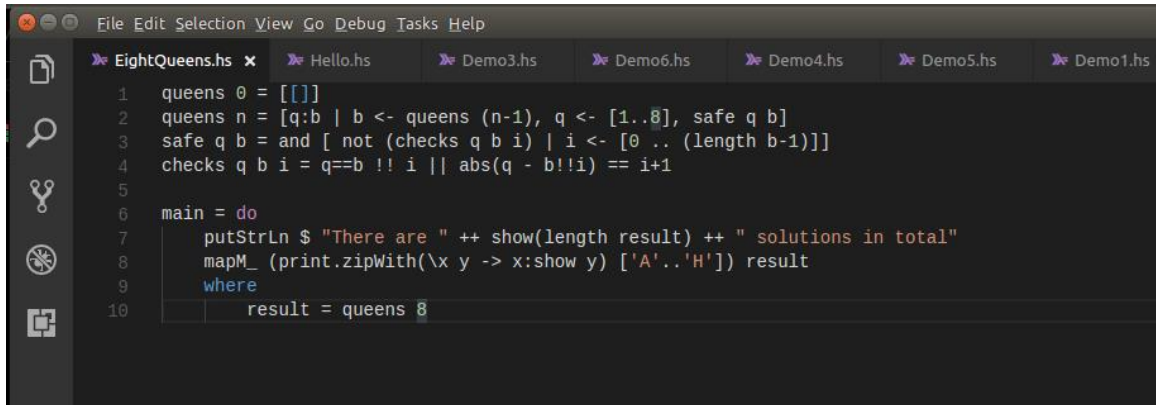


作业 1: Eight Queens

一: Source Code:



```
File Edit Selection View Go Debug Tasks Help
EightQueens.hs x Hello.hs Demo3.hs Demo6.hs Demo4.hs Demo5.hs Demo1.hs
1 queens 0 = [[]]
2 queens n = [q:b | b <- queens (n-1), q <- [1..8], safe q b]
3 safe q b = and [ not (checks q b i) | i <- [0 .. (length b-1)]]
4 checks q b i = q==b !! i || abs(q - b!!i) == i+1
5
6 main = do
7     putStrLn $ "There are " ++ show(length result) ++ " solutions in total"
8     mapM_ (print.zipWith(\x y -> x:show y) ['A'..'H']) result
9     where
10        result = queens 8
```

详细代码请查看附件 1

二：实验结果：

```
File Edit Selection View Go Debug Tasks Help

EightQueens.hs x Hello.hs Demo3.hs Demo6.hs Demo4.hs

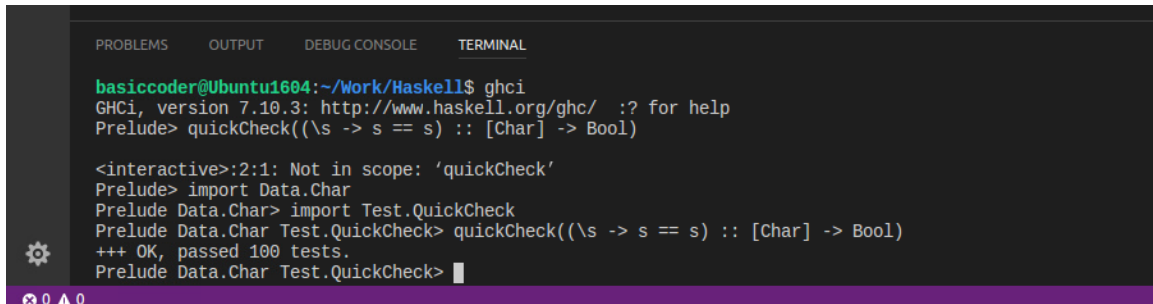
1 queens 0 = [[]]
2 queens n = [q:b | b <- queens (n-1), q <- [1..8], safe q b]
3 safe q b = and [ not (checks q b i) | i <- [0 .. (length b-1)]]
4 checks q b i = q==b !! i || abs(q - b!!i) == i+1
5

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

basiccoder@Ubuntu1604:~/Work/Haskell$ ghc EightQueens.hs
[1 of 1] Compiling Main (EightQueens.hs, EightQueens.o)
Linking EightQueens ...
basiccoder@Ubuntu1604:~/Work/Haskell$ ./EightQueens
There are 92 solutions in total
["A4","B2","C7","D3","E6","F8","G5","H1"]
["A5","B2","C4","D7","E3","F8","G6","H1"]
["A3","B5","C2","D8","E6","F4","G7","H1"]
["A3","B6","C4","D2","E8","F5","G7","H1"]
["A5","B7","C1","D3","E8","F6","G4","H2"]
["A4","B6","C8","D3","E1","F7","G5","H2"]
["A3","B6","C8","D1","E4","F7","G5","H2"]
["A5","B3","C8","D4","E7","F1","G6","H2"]
["A5","B7","C4","D1","E3","F8","G6","H2"]
["A4","B1","C5","D8","E6","F3","G7","H2"]
["A3","B6","C4","D1","E8","F5","G7","H2"]
["A4","B7","C5","D3","E1","F6","G8","H2"]
["A6","B4","C2","D8","E5","F7","G1","H3"]
["A6","B4","C7","D1","E8","F2","G5","H3"]
["A1","B7","C4","D6","E8","F2","G5","H3"]
["A6","B8","C2","D4","E1","F7","G5","H3"]
["A6","B2","C7","D1","E4","F8","G5","H3"]
["A4","B7","C1","D8","E5","F2","G6","H3"]
["A5","B8","C4","D1","E7","F2","G6","H3"]
["A4","B8","C1","D5","E7","F2","G6","H3"]
["A2","B7","C5","D8","E1","F4","G6","H3"]
["A1","B7","C5","D8","E2","F4","G6","H3"]
["A2","B5","C7","D4","E1","F8","G6","H3"]
["A4","B2","C7","D5","E1","F8","G6","H3"]
["A5","B7","C1","D4","E2","F8","G6","H3"]
["A6","B4","C1","D5","E8","F2","G7","H3"]
["A5","B1","C4","D6","E8","F2","G7","H3"]
["A5","B2","C6","D1","E7","F4","G8","H3"]
["A6","B3","C7","D2","E8","F5","G1","H4"]
["A2","B7","C3","D6","E8","F5","G1","H4"]
["A7","B3","C1","D6","E8","F5","G2","H4"]
["A5","B1","C8","D6","E3","F7","G2","H4"]
["A1","B5","C8","D6","E3","F7","G2","H4"]
["A3","B6","C8","D1","E5","F7","G2","H4"]
["A6","B3","C1","D7","E5","F8","G2","H4"]
["A7","B5","C3","D1","E6","F8","G2","H4"]
["A7","B3","C8","D2","E5","F1","G6","H4"]
["A5","B3","C1","D7","E2","F8","G6","H4"]
["A2","B5","C7","D1","E3","F8","G6","H4"]
["A3","B6","C2","D5","E8","F1","G7","H4"]
["A6","B1","C5","D2","E8","F3","G7","H4"]
["A8","B3","C1","D6","E2","F5","G7","H4"]
["A2","B8","C6","D1","E3","F5","G7","H4"]
["A5","B7","C2","D6","E3","F1","G8","H4"]
["A3","B6","C2","D7","E5","F1","G8","H4"]
["A6","B2","C7","D1","E3","F5","G8","H4"]
["A3","B7","C2","D8","E6","F4","G1","H5"]
```

作业 2: QuickCheck

一. 实验过程:



```
basiccoder@Ubuntu1604:~/Work/Haskell$ ghci
GHCi, version 7.10.3: http://www.haskell.org/ghc/ :? for help
Prelude> quickCheck((\s -> s == s) :: [Char] -> Bool)

<interactive>:2:1: Not in scope: 'quickCheck'
Prelude> import Data.Char
Prelude Data.Char> import Test.QuickCheck
Prelude Data.Char Test.QuickCheck> quickCheck((\s -> s == s) :: [Char] -> Bool)
+++ OK, passed 100 tests.
Prelude Data.Char Test.QuickCheck> 
```

附件:

[1] EightQueens.hs