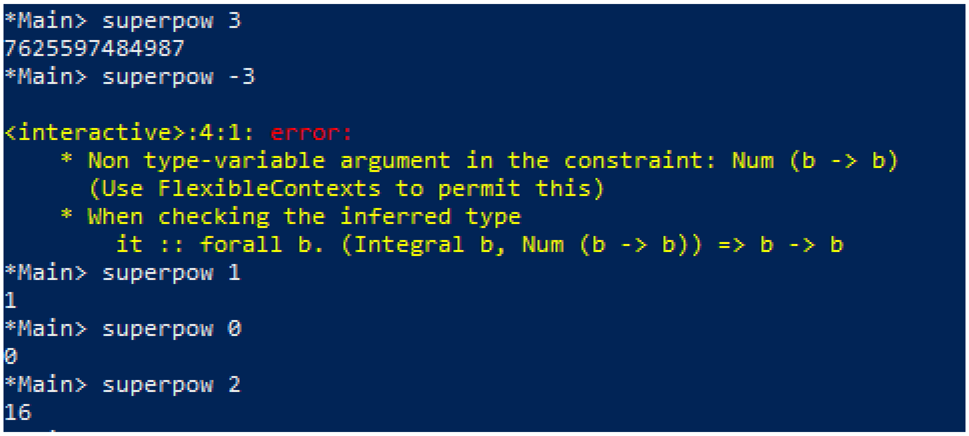
Stuart Harley

# Assignment 2 Evidence Document

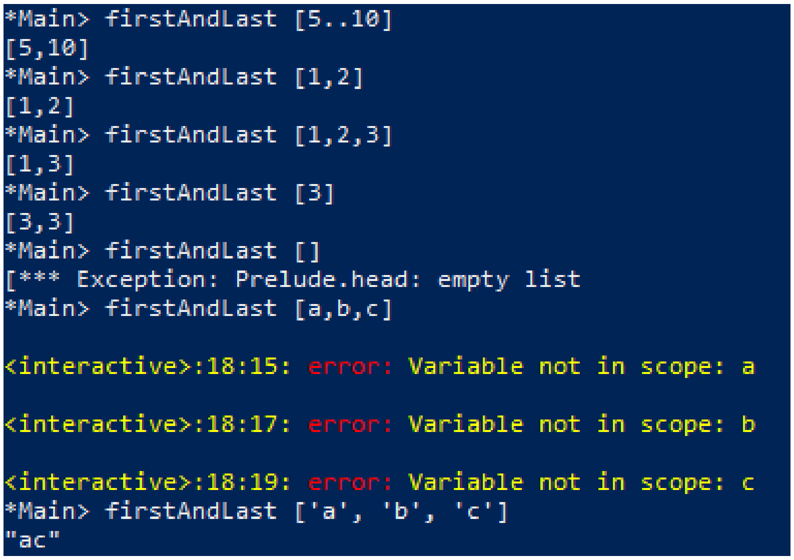
## Problem 1)



```
*Main> superpow 3
7625597484987
*Main> superpow -3

<interactive>:4:1: error:
    * Non type-variable argument in the constraint: Num (b -> b)
      (Use FlexibleContexts to permit this)
    * When checking the inferred type
        it :: forall b. (Integral b, Num (b -> b)) => b -> b
*Main> superpow 1
1
*Main> superpow 0
0
*Main> superpow 2
16
```

## Problem 2)



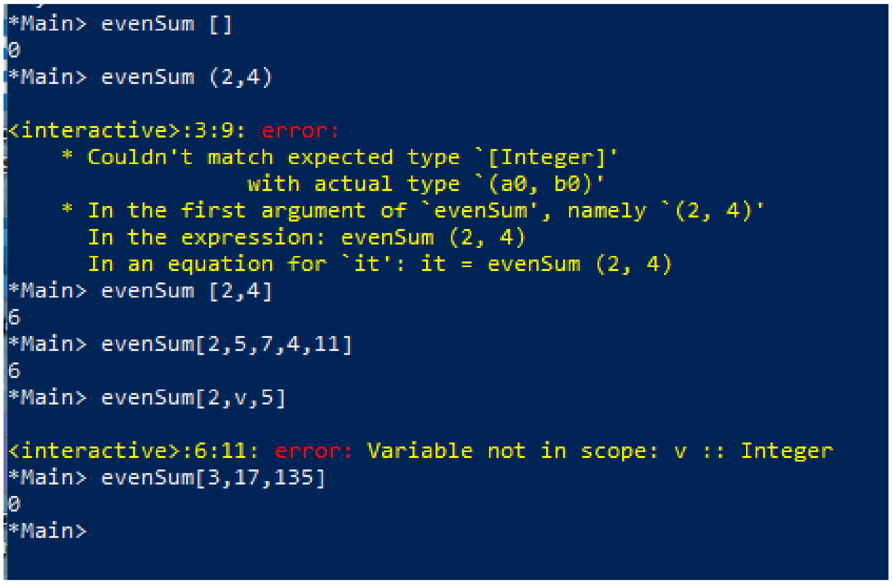
```
*Main> firstAndLast [5..10]
[5,10]
*Main> firstAndLast [1,2]
[1,2]
*Main> firstAndLast [1,2,3]
[1,3]
*Main> firstAndLast [3]
[3,3]
*Main> firstAndLast []
[*** Exception: Prelude.head: empty list
*Main> firstAndLast [a,b,c]

<interactive>:18:15: error: Variable not in scope: a

<interactive>:18:17: error: Variable not in scope: b

<interactive>:18:19: error: Variable not in scope: c
*Main> firstAndLast ['a', 'b', 'c']
"ac"
```

Problem 3)

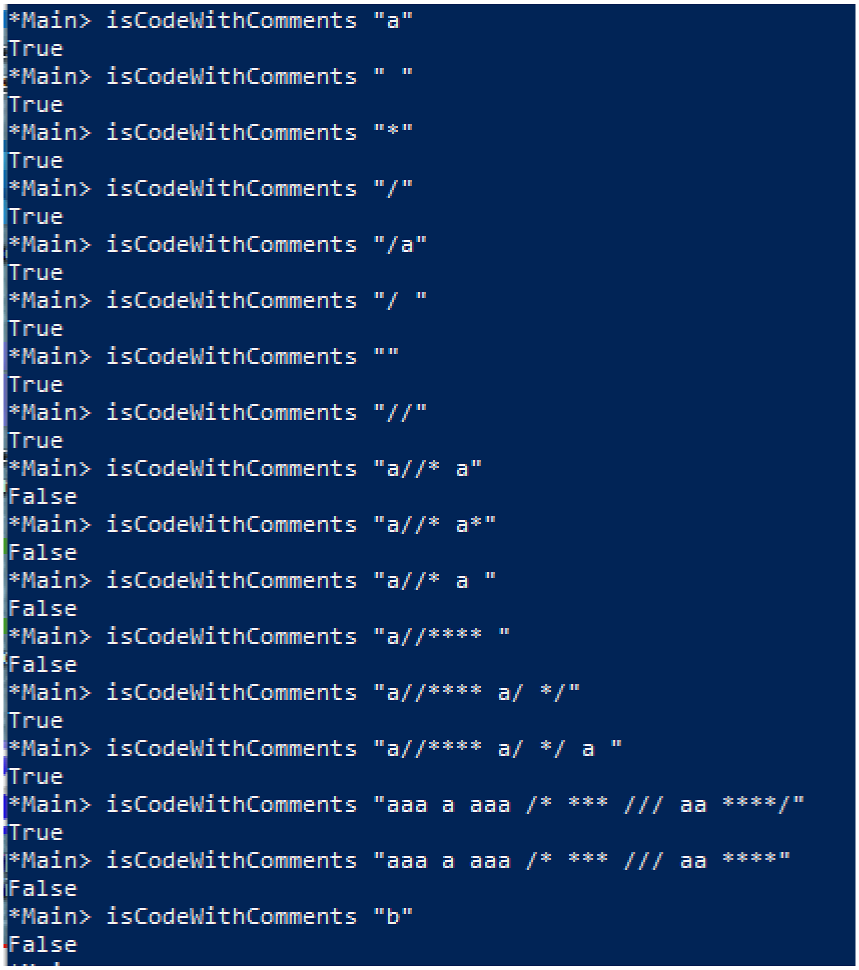


```
*Main> evenSum []
0
*Main> evenSum (2,4)

<interactive>:3:9: error:
    * Couldn't match expected type `[Integer]'
                  with actual type `(a0, b0)'
    * In the first argument of `evenSum', namely `(2, 4)'
      In the expression: evenSum (2, 4)
      In an equation for `it': it = evenSum (2, 4)
*Main> evenSum [2,4]
6
*Main> evenSum[2,5,7,4,11]
6
*Main> evenSum[2,v,5]

<interactive>:6:11: error: Variable not in scope: v :: Integer
*Main> evenSum[3,17,135]
0
*Main>
```

Problem 4)



```
*Main> isCodeWithComments "a"
True
*Main> isCodeWithComments " "
True
*Main> isCodeWithComments "*"
True
*Main> isCodeWithComments "/"
True
*Main> isCodeWithComments "/a"
True
*Main> isCodeWithComments "/ "
True
*Main> isCodeWithComments ""
True
*Main> isCodeWithComments "//"
True
*Main> isCodeWithComments "a//* a"
False
*Main> isCodeWithComments "a//* a*"
False
*Main> isCodeWithComments "a//* a "
False
*Main> isCodeWithComments "a//**** "
False
*Main> isCodeWithComments "a//**** a/ */"
True
*Main> isCodeWithComments "a//**** a/ */ a "
True
*Main> isCodeWithComments "aaa a aaa /* *** /// aa ****/"
True
*Main> isCodeWithComments "aaa a aaa /* *** /// aa ****"
False
*Main> isCodeWithComments "b"
False
```