-- Zayadur Khan CIS-252 Assignment 10

-- helper functions from April 18 lecture

getInteger :: IO Integer

getInteger = do line <- getLine

return (read line :: Integer)

getFloat :: IO Float

getFloat = do line <- getLine

return (read line :: Float)

-- stats prompts user to enter 3 numbers, reads in 3 Floats,

-- then returns a pair containing the mean and median of

-- the three numbers

stats :: IO (Float, Float)

stats = do putStrLn "Enter 3 numbers: "

x <- getFloat

y <- getFloat

z <- getFloat

return ((x+y+z)/3, (maximum [m | m <- [x,y,z], m /= maximum [x,y,z] && m /= minimum [x,y,z]]))

-- printVert cs displays the string cs vertically

printVert :: String -> IO Int

printVert cs = do sequence [putStrLn [x] | x <- cs]

return (length cs)

-- displayWords prompts the user to enter line of text,

-- reads that text, and displays each word in its own line

displayWords = do putStr "Enter text: "

text <- getLine

sequence\_ [putStrLn x | x <- (words text)]

-- displayWords2 does the same thing as displayWords,

-- but it numbers each word starting from 1

displayWords2 :: IO ()

displayWords2 = do putStr "Enter text: "

text <- getLine

sequence\_ [putStrLn (show y ++ ". " ++ x) | (x,y) <- zip (words text) [1..]]

-- nonzeros reads a series of Integers,

-- stopping only when it reads a 0,

-- then it returns the list of nonzero values read

nonzeros :: IO [Integer]

nonzeros = do int <- getInteger

if (int == 0)

then return []

else do lst <- nonzeros

return (int : lst)

-- posAndNegs reads in unbounded series of Integers,

-- stopping only when it reads value 0,

-- then displays message indicating:

-- number of nonzero numbers,

-- number of positive numbers,

-- smallest negative number entered

posAndNegs :: IO ()

posAndNegs = do putStrLn "Enter series: "

ints <- nonzeros

putStr "Frequency of nonzeros: "

print (length ints)

putStr "Frequency of positives: "

print (length [x | x <- ints, x > 0])

if(minimum ints > 0)

then putStrLn "No negative numbers detected"

else do putStr "Smallest negative number: "

print (minimum ints)