```
--approx x : An infinite list of approximations which converge on the square
--root of a negitive float 'x'
approx :: Float -> [Float]
approx x = 1.0 : [ (prev + x / prev) / 2.0 | prev <- approx x ]

--squareRoot x : An estimate of the square root of non negitive float 'x'
squareRoot :: Float -> Float
squareRoot x = converge (approx x) 0.0001

--converge xs diff : The first item in infinite list 'xs' which differs to
--                   the previous item by less than 'diff'
converge :: [Float] -> Float -> Float
converge (x:y:xs) diff
  | (abs (x -y)) >= diff = converge (y:xs) diff
  | otherwise = y

--iSquareRoot x : An integer estimate of the square root of integer 'x'
iSquareRoot :: Integer -> Integer
iSquareRoot x = floor ( squareRoot ( fromIntegral x) )

--primes : The infinite list of prime numbers
primes :: [Integer]
primes = 2:[p | p <- [3,5..], null (primeFactors p) ]
--primeFactors : The prime factors of an integer
primeFactors :: Integer -> [Integer]
primeFactors x = [f | f <- takeWhile (<= iSquareRoot x) primes, mod x f ==0 ]
```