1 Problem

Using *names.txt* (right click and 'Save Link/Target As...'), a 46K text file containing over five-thousand first names, begin by sorting it into alphabetical order. Then working out the alphabetical value for each name, multiply this value by its alphabetical position in the list to obtain a name score.

For example, when the list is sorted into alphabetical order, COLIN, which is worth $3 + 15 + 12 + 9 + 14 = 53$, is the 938^{th} name in the list. So, COLIN would obtain a score of $938 \times 53 = 49714$.

What is the total of all the name scores in the file?

2 Solution

```
import Data.Maybe
import Data.List
import Data.Char

split :: Char → String → [String]
split = unfoldr ∘ split'

split' :: Char → String → Maybe (String, String)
split' c l
  | null l = Nothing
  | otherwise = Just (h, drop 1 t)
  where (h, t) = span (≢ c) l

parseNameFile :: String → IO [String]
parseNameFile fname = do
  rawText ← readFile fname
  let fields = split ',' rawText
      names = map (reverse ∘ tail ∘ reverse ∘ tail) fields
  return $ sort names

nameVal :: String → Int
nameVal st = sum $ map (λz → ord z − 64) st

scores :: [String] → Int
scores nf = sum $ zipWith (λx y → x ∗ (nameVal y)) [1 .. (length nf)] nf

main = do
  names ← parseNameFile "names.txt"
  let sc = scores names
  putStrLn $ "The sum of all scores is " ++ show sc ++ "."
```

3 Result

```
runhaskell problem22.lhs
The sum of all scores is 871198282.
```