Homework Assignment 2 – Solutions

October 20, 2013

We load the 2 texts. We have to load the text in the file first because `ghci` wipes all bindings every time a file is loaded.

```
ghci 1> :l textBrown
```

```
ghci 2> let text = "Pierre Vinken , 61 years old , will join the board " ++
          "as a nonexecutive director Nov. 29 .\nMr. Vinken " ++
          "is chairman of Elsevier N.V. , the Dutch publishing group ."
```

1 Lines, words, checking for words in lines

A. Split this text into lines (on "\n"), extract the first sentence, then the second sentence, then print the list of words for the two sentences

```
ghci 3> let ls_text = lines text
```

```
ghci 4> ls_text
   ["Pierre Vinken , 61 years old , will join the board as a nonexecutive director Nov. 29 .",
   " Mr. Vinken is chairman of Elsevier N.V. , the Dutch publishing group ."]
```

```
ghci 5> ls_text !! 0
   "Pierre Vinken , 61 years old , will join the board as a nonexecutive director Nov. 29 ."
```

```
ghci 6> ls_text !! 1
   " Mr. Vinken is chairman of Elsevier N.V. , the Dutch publishing group ."
```

```
ghci 7> let ws_s1 = words (ls_text !! 0)
ghci 8> ws_s1
   ["Pierre","Vinken",",","61","years","old",",","will","join","the","board","as",
   "a","nonexecutive","director","Nov.","29","."]
```

```
ghci 9> let ws_s2 = words (ls_text !! 1)
```

```
ghci 10> ws_s2
   ["Mr.","Vinken","is","chairman","of","Elsevier","N.V.",",","the","Dutch",
   "publishing","group","."]
```

B. Check if the word “Vinken” is one of the words in the first sentence and separately, if it’s one of the words in the second sentence

```
ghci 11> "Vinken" ∈ ws_s1
   True
```

```
ghci 12> "Vinken" ∈ ws_s2
   True
```

C. Check if the word “chairman” is **not** one of the words in the first sentence and separately, if it’s **not** one of the words in the second sentence

```
ghci 13> ¬ ("chairman" ∈ ws_s1)
   True
```

```
ghci 14> ¬ ("chairman" ∈ ws_s2)
   False
```

```
ghci 15> "chairman" ∉ ws_s1
   True
```

```
ghci 16> "chairman" ∉ ws_s2
   False
```

2 Word sets

A. Find a function in the module *Data.List* that enables you to extract the set of words in each of the 2 sentences – and print the 2 sets of words

```
ghci 17> import Data.List (nub)
```

```
ghci 18> nub ws_s1
    ["Pierre","Vinken",",","61","years","old","will","join","the","board","as","a",
    "nonexecutive","director","Nov.","29","."]
```

```
ghci 19> nub ws_s2
    ["Mr.","Vinken","is","chairman","of","Elsevier","N.V.",",","the","Dutch",
    "publishing","group","."]
```

B. Find a way to extract the set of words in both sentences (no duplicates) and print it

```
ghci 20> map words ls_text
    [["Pierre","Vinken",",","61","years","old",",","will","join","the","board",
    "as","a","nonexecutive","director","Nov.","29","."],["Mr.","Vinken","is",
    "chairman","of","Elsevier","N.V.",",","the","Dutch","publishing","group","."]]
```

```
ghci 21> let ws_text = concat (map words ls_text)
```

```
ghci 22> ws_text
    ["Pierre","Vinken",",","61","years","old",",","will","join","the","board","as",
    "a","nonexecutive","director","Nov.","29",".","Mr.","Vinken","is","chairman",
    "of","Elsevier","N.V.",",","the","Dutch","publishing","group","."]
```

```
ghci 23> nub ws_text
    ["Pierre","Vinken",",","61","years","old","will","join","the","board","as",
    "a","nonexecutive","director","Nov.","29",".","Mr.","is","chairman","of",
    "Elsevier","N.V.","Dutch","publishing","group"]
```

C. Find a way to count how many comma tokens occur in the text; do the same for the definite article "the":

```
ghci 24> let {countToken :: (Eq a, Num b) ⇒ a → [a] → b;
              countToken _ [] = 0;
              countToken x (y : ys)
                | x ≡ y = 1 + countToken x ys
                | otherwise = countToken x ys
              }
```

```
ghci 25> countToken "," ws_text
    3
```

```
ghci 26> countToken "the" ws_text
    2
```

```
ghci 27> countToken "Vinken" ws_text
    2
```

```
ghci 28> countToken "," (nub ws_text)
    1
```

```
ghci 29> countToken "the" (nub ws_text)
    1
```

D. Find a way to count the tokens for every word that occurs in the text and print the resulting counts:

```
ghci 30> let {countItemsInList :: (Eq a, Num b) ⇒ [a] → [a] → [(a, b)];
         countItemsInList [ ] _ = [ ];
         countItemsInList (x : xs) ys =
           (x, countToken x ys) : countItemsInList xs ys}
```

```
ghci 31> let {tokenCounts :: (Eq a, Num b) ⇒ [a] → [(a, b)];
         tokenCounts xs = countItemsInList (nub xs) xs}
```

```
ghci 32> tokenCounts ws_text
    [("Pierre",1),("Vinken",2),(",",3),("61",1),("years",1),("old",1),("will",1),
    ("join",1),("the",2),("board",1),("as",1),("a",1),("nonexecutive",1),("director",
    1),("Nov.",1),("29",1),(".",2),("Mr.",1),("is",1),("chairman",1),("of",1),
    ("Elsevier",1),("N.V.",1),("Dutch",1),("publishing",1),("group",1)]
```

E. Using the same functions, count the word/tag pairs in the following text from the Brown corpus:

```
ghci 33> map words (lines textBrown)
    [["The/at","Fulton/np-tl","County/nn-tl","Grand/jj-tl","Jury/nn-tl",
    "said/vbd","Friday/nr","an/at","investigation/nn","of/in","Atlanta’s/np$",
    "recent/jj","primary/nn","election/nn","produced/vbd","no/at","evidence/nn",
    "that/cs","any/dti","irregularities/nns","took/vbd","place/nn","./."],
    ["The/at","jury/nn","further/rbr","said/vbd","in/in","term-end/nn",
    "presentments/nns","that/cs","the/at","City/nn-tl","Executive/jj-tl",
    "Committee/nn-tl",",/,","which/wdt","had/hvd","over-all/jj","charge/nn",
    "of/in","the/at","election/nn",",/,","deserves/vbz","the/at","praise/nn",
    "and/cc","thanks/nns","of/in","the/at","City/nn-tl","of/in-tl",
    "Atlanta/np-tl","for/in","the/at","manner/nn","in/in","which/wdt",
    "the/at","election/nn","was/bedz","conducted/vbn","./."],["The/at",
    "September-October/np","term/nn","jury/nn","had/hvd","been/ben","charged/vbn",
    "by/in","Fulton/np-tl","Superior/jj-tl","Court/nn-tl","Judge/nn-tl",
    "Durwood/np","Pye/np","to/to","investigate/vb","reports/nns","of/in",
    "possible/jj","irregularities/nns","in/in","the/at","hard-fought/jj",
    "primary/nn","which/wdt","was/bedz","won/vbn","by/in","Mayor-nominate/nn-tl",
    "Ivan/np","Allen/np","Jr./np","./."],["Only/rb","a/at","relative/jj",
    "handful/nn","of/in","such/jj","reports/nns","was/bedz","received/vbn",",/,",
    "the/at","jury/nn","said/vbd",",/,","considering/in","the/at","widespread/jj",
    "interest/nn","in/in","the/at","election/nn",",/,","the/at","number/nn",
    "of/in","voters/nns","and/cc","the/at","size/nn","of/in","this/dt","city/nn",
    "./."],["The/at","jury/nn","said/vbd","it/pps","did/dod","find/vb","that/cs",
    "many/ap","of/in","Georgia’s/np$","registration/nn","and/cc","election/nn",
    "laws/nns","are/ber","outmoded/jj","or/cc","inadequate/jj","and/cc",
    "often/rb","ambiguous/jj","./."],["It/pps","recommended/vbd","that/cs",
    "Fulton/np","legislators/nns","act/vb","to/to","have/hv","these/dts",
    "laws/nns","studied/vbn","and/cc","revised/vbn","to/in","the/at","end/nn",
    "of/in","modernizing/vbg","and/cc","improving/vbg","them/ppo","./."]]
```

```
ghci 34> let ws_textBrown = concat (map words (lines textBrown))
ghci 35> tokenCounts ws_textBrown
    [("The/at",4),("Fulton/np-tl",2),("County/nn-tl",1),("Grand/jj-tl",1),
    ("Jury/nn-tl",1),("said/vbd",4),("Friday/nr",1),("an/at",1),("investigation/nn",
    1),("of/in",9),("Atlanta's/np$",1),("recent/jj",1),("primary/nn",2),
    ("election/nn",5),("produced/vbd",1),("no/at",1),("evidence/nn",1),("that/cs",
    4),("any/dti",1),("irregularities/nns",2),("took/vbd",1),("place/nn",1),
    ("./.",6),("jury/nn",4),("further/rbr",1),("in/in",4),("term-end/nn",1),
    ("presentments/nns",1),("the/at",13),("City/nn-tl",2),("Executive/jj-tl",1),
    ("Committee/nn-tl",1),(",/,",5),("which/wdt",3),("had/hvd",2),("over-all/jj",1),
    ("charge/nn",1),("deserves/vbz",1),("praise/nn",1),("and/cc",6),("thanks/nns",
    1),("of/in-tl",1),("Atlanta/np-tl",1),("for/in",1),("manner/nn",1),("was/bedz",
    3),("conducted/vbn",1),("September-October/np",1),("term/nn",1),("been/ben",
    1),("charged/vbn",1),("by/in",2),("Superior/jj-tl",1),("Court/nn-tl",1),
    ("Judge/nn-tl",1),("Durwood/np",1),("Pye/np",1),("to/to",2),("investigate/vb",
    1),("reports/nns",2),("possible/jj",1),("hard-fought/jj",1),("won/vbn",
    1),("Mayor-nominate/nn-tl",1),("Ivan/np",1),("Allen/np",1),("Jr./np",1),
    ("Only/rb",1),("a/at",1),("relative/jj",1),("handful/nn",1),("such/jj",1),
    ("received/vbn",1),("considering/in",1),("widespread/jj",1),("interest/nn",
    1),("number/nn",1),("voters/nns",1),("size/nn",1),("this/dt",1),("city/nn",
    1),("it/pps",1),("did/dod",1),("find/vb",1),("many/ap",1),("Georgia's/np$",
    1),("registration/nn",1),("laws/nns",2),("are/ber",1),("outmoded/jj",1),
    ("or/cc",1),("inadequate/jj",1),("often/rb",1),("ambiguous/jj",1),("It/pps",
    1),("recommended/vbd",1),("Fulton/np",1),("legislators/nns",1),("act/vb",1),
    ("have/hv",1),("these/dts",1),("studied/vbn",1),("revised/vbn",1),("to/in",1),
    ("end/nn",1),("modernizing/vbg",1),("improving/vbg",1),("them/ppo",1)]
```