

Burt Rosenberg

## Problem Set 4

OUT: 16 SEPTEMBER, 1993

DUE: 24 SEPTEMBER, 1993

### Goals

Practice using pointers, linked lists and text files.

### Reading Assignment

Read Chapter 3 from *Algorithms* by Robert Sedgewick.

### Programming Assignment

Write a program which keeps a linked list of the *unique* words in a text file, along with a count as to how many times each unique word appeared in the text. For instance, the text “Romeo, Romeo! wherefore art thou Romeo?” has 4 unique words, “romeo” with a count of 3, “wherefore”, “art” and “thou” each with a count of 1.

The output should be the total number of words, the number of unique words, and a list of the unique words with their count. To make sure your code really works, you should take a small file and watch what it does with each word read. Below I give an example session for a very small file. Hand in a file which is a record of this session.

As a guide to the solution, I suggest you work as follows:

1. Change the type of the nodes in the list to also have a field *count* of type integer.
2. Write a function to check equality of strings.
3. Write a function returning a pointer to a node in the list or *nil* which searches a list for a match with a given string.
4. Combine the above to implement the unique words program.

## Testing

In the given example, here is the sequence of lists. The variable  $p$  is the list's anchor.

1. First, there is an empty list, assume a dummy header node carrying a nonsense string  $\Omega$  with count 0,

$$p \longrightarrow (\Omega, 0)$$

2. Next, the word "romeo" is inserted:

$$p \longrightarrow (\Omega, 0) \longrightarrow (\text{romeo}, 1)$$

3. Again, "romeo":

$$p \longrightarrow (\Omega, 0) \longrightarrow (\text{romeo}, 2)$$

4. Now, "wherefore" is added to the front:

$$p \longrightarrow (\Omega, 0) \longrightarrow (\text{wherefore}, 1) \longrightarrow (\text{romeo}, 2)$$

5. And "art" and "thou" (skipping ahead):

$$p \longrightarrow (\Omega, 0) \longrightarrow (\text{thou}, 1) \longrightarrow (\text{art}, 1) \longrightarrow (\text{wherefore}, 1) \longrightarrow (\text{romeo}, 2)$$

6. Finally, another "romeo":

$$p \longrightarrow (\Omega, 0) \longrightarrow (\text{thou}, 1) \longrightarrow (\text{art}, 1) \longrightarrow (\text{wherefore}, 1) \longrightarrow (\text{romeo}, 3)$$

Have your code print out after every word, to make sure this sequence is what you get. The final output for this test is:

```
6 words in the file,
4 unique words. They are:
  thou      1
  art       1
  wherefore 1
  romeo     3
```