Answer Set 2

OUT: 27 SEPTEMBER, 1994

----------ANSWER2.PAS----------

```
[ENVIRONMENT,INHERIT('mystring2')]

module answer2(input,output) ;

(* Answer to problem set 2,
   a module for lists of strings
   (c) Burton Rosenberg 1994
   Univ. of Miami
*)

type
   MyList = ^MyListElement ;
   MyListElement = record
                   theMs : MyString ;
                   count : integer ;
                   next : MyList ;
               end ;
   MyListCoupon = MyList ;

procedure create_MyList(var ml : MyList ) ;
begin
   ml := NIL ;
end ;

procedure print_MyList( ml : MyList ) ;
begin
   while (ml<>NIL) do begin
       write(ml^.count:3,' ') ;
       writeln_MyString( ml^.theMs ) ;
       ml := ml^.next
   end
end ;
```

```
procedure add_MyList( var ml: Mylist; data: MyString; count: integer ) ;
var
   p : MyList ;
begin
   new(p) ;
   p^.next := ml ;
   ml := p ;
   ml^.count := count ;
   copy_MyString( ml^.theMs, data )
end ;

function find_MyList( ml: MyList; toFind: MyString ) : MyListCoupon ;
var p : MyList ;
begin
   p := NIL ;
   (*L.I. toFind not found on list up till predecessor of ml^.
          ml^ to check *)
   while ( p<>ml ) do
      if compareNC_MyString(ml^.theMs, toFind) then p := ml
      else ml := ml^.next ;
   find_MyList := p ;
end ;

procedure Increment_MyList( ml: Mylist; coup: MyListCoupon ) ;
begin  coup^.count := coup^.count+1   end ;

procedure Statistics_MyList( ml: MyList; var len: integer; var sum: integer ) ;
begin
   len := 0 ;
   sum := 0 ;
   (* L.I. len is number of list items up till, not including ml^,
           sum is the sum of count fields for list items up till,
           not including ml^.count. *)
   while (ml<>NIL) do begin
      len := len+1 ;
      sum := sum + ml^.count ;
      ml := ml^.next
   end
end ;

end.
```

```
-----------------------DRIVER2.PAS-------------------------

[INHERIT('MYSTRING2','ANSWER2')]
program driver1( input, output ) ;

(*  driver program to test answer2.pas
   (c) Burton Rosenberg, U. of Miami 1994
*)
var
   ml : MyList  ;
   aLine, aWord : MyString ;
   p : MyListCoupon ;
   i : integer ;
   numWords, totalCount : integer ;

begin

   create_MyList( ml ) ;
   i := readln_MyString( aLine ) ;
   (* aLine contains a line of test to process, all
      previous lines have been processed into mylist ml *)
   while ( i<>MYEOF ) do begin

      parse_MyString( aLine, aWord ) ;
      (* aWord has a word from the current line to process *)
      while ( length_MyString( aWord ) > 0 ) do begin

         (* if word not on list, add it, else increment the
            word's count *)
         p := find_MyList( ml, aWord ) ;
         if (p=NIL) then
            add_MyList( ml, aWord, 1 )
         else
            increment_MyList( ml, p ) ;
         parse_MyString( aLine, aWord ) ;

      end {while length ...} ;
      i := readln_MyString( aLine ) ;
   end { while not MYEOF } ;

   (* calculate and print results *)
   Statistics_MyList( ml, numWords, totalCount ) ;
   writeln(totalCount:5,' words in the file,') ;
   writeln(numWords:5,' unique words. They are:') ;
```

```
   print_MyList( ml ) ;
end .
```

Sample Runs

```
$ pascal mystring2
$ pascal answer2
$ pascal driver2
$ link driver2,answer2,mystring2
$ run driver2
Romeo, Romeo!   wherefore
art    thou,    Romeo   ?
[Exit]
    6 words in the file,
    4 unique words. They are:
  1 thou
  1 art
  1 wherefore
  3 Romeo
$
```