**Exercise 6-1.** Our version of getword does not properly handle underscores, string constants, comments, or preprocessor control lines. Write a better version.

**Exercise 6-2.** Write a program that reads a C program and prints in alphabetical order each group of variable names that are identical in the first 6 characters, but different somewhere thereafter. Don't count words within strings and comments. Make 6 a parameter that can be set from the command line.

**Exercise 6-3.** Write a cross-referencer that prints a list of all words in a document, and for each word, a list of the line numbers on which it occurs. Remove noise words like ``the,'' ``and,'' and so on.

**Exercise 6-4.** Write a program that prints the distinct words in its input sorted into decreasing order of frequency of occurrence. Precede each word by its count.

**Exercise 6-5.** Write a function undef that will remove a name and definition from the table maintained by lookup and install.

**Exercise 6-6.** Implement a simple version of the #define processor (i.e., no arguments) suitable for use with C programs, based on the routines of this section. You may also find getch and ungetch helpful.