21.1 (*Perform set operations on hash sets*) Create two linked hash sets {`"George"`, `"Jim"`, `"John"`, `"Blake"`, `"Kevin"`, `"Michael"`} and {`"George"`, `"Katie"`, `"Kevin"`, `"Michelle"`, `"Ryan"`} and find their union, difference, and intersection. (You can clone the sets to preserve the original sets from being changed by these set methods.)

21.2 (*Display nonduplicate words in ascending order*) Write a program that reads words from a text file and displays all the nonduplicate words in ascending order. The text file is passed as a command-line argument.

****21.3** (*Count the keywords in Java source code*) Revise the program in Listing 21.7. If a keyword is in a comment or in a string, don't count it. Pass the Java file name from the command line. Assume that the Java source code is correct and line comments and paragraph comments do not overlap.

***21.4** (*Count consonants and vowels*) Write a program that prompts the user to enter a text file name and displays the number of vowels and consonants in the file. Use a set to store the vowels `A`, `E`, `I`, `O`, and `U`.

****21.8** (*Count the occurrences of words in a text file*) Rewrite Listing 21.9 to read the text from a text file. The text file is passed as a command-line argument. Words are delimited by whitespace characters, punctuation marks (`,;.:?`), quotation marks (`'"`), and parentheses. Count words in case-insensitive fashion (e.g., consider `Good` and `good` to be the same word). The words must start with a letter. Display the output in alphabetical order of words, with each word preceded by its occurrence count.