Homework #5

Due: June 5, 2014 at 11:55pm (Note different time!) **Points:** 100

This homework has you implement a cross-reference generator. This program takes a text file as input, and produces a list of words and the line numbers on which each word appears.

1. (*35 points*) The program “xref.py” (available in the *programs* area of SmartSite and the backup web site) is to read in a file and split each line into words. Here, a “word” is a maximal string of upper case letters, lower case letters, digits, and underscores “_” (in other words, the characters that are legal in Python identifiers, *except* that a word may begin with a digit). It then prints the number of times each word occurs in the file. However, when you try to run it, you will see it is missing two routines: `getfname()`, which prompts the user for a file name, and `openfile(`*fname*`)`, which opens the file *fname* for reading.

   You are to add these. The specification for each follows:

   - `getfname()` takes no arguments. It prompts the user to enter the name of a file, and returns either that name or the distinguished value `None` (no quotation marks; it is not a string). If the user enters an end-of-file (control-d on most systems), simply return the distinguished value `None`; if any other error on input occurs, print the error message “Error reading file name” and return distinguished value `None`.
   - `openfile(`*fname*`)` takes one argument, a file name *fname*. It opens the file for reading and returns the file handle. If the open fails, the routine prints the error message “Could not open file!” and returns the distinguished value `None`, as above.

   When the program works and is run on the input file “alice.txt”, the first few lines of output will look like this:

   ```
   drink                        1
   waistcoat                    1
   perhaps                      1
   question                     1
   June                         1
   conversation                 1
   on                           10
   see                          8
   going                        2
   ```

   The words listed may vary, because of the ways dictionaries are implemented.

   Submit. Name your file “xref1.py” and submit it to the Homework #5 area for this class on SmartSite.

2. (*10 points*) Modify your program in “xref1.py” to print the words in sorted order. The input is to be as above. The output is to be formatted as above, but the words are to be sorted. So, for the file “alice.txt”, the first few lines of the output should be:

   ```
   0                            1
   11                           1
   1994                         1
   20                           1
   2008                         1
   2011                         1
   25                           1
   3                            1
   A                            1
   ```

```
ADVENTURES              2
ALICE                   2
ASCII                   1
Adventures              2
Alice                   23
```

The words listed above should be what is in your output, as should the numbers.

Submit. Name your file "xref1s.py" and submit it to the Homework #5 area for this class on SmartSite.

Hint: Use `sorted(d)` to sort the dictionary `d` based on the keys.

3. (*55 points*) Now you will build the cross-reference part. Begin with "xref1s.py". Instead of the number of times you see each word, construct a list of line numbers on which the word occurs.

   Input. Read in the name of a file to process. If there is an error in opening the file, print "Error reading file" and quit.

   Output. Print out a list of words in the file, one word per line, followed by a comma-separated list of line numbers that word occurs in the file. So, for the file "alice.txt", the *last* few lines of the output should be:

```
words                   101, 170
world                   68
worth                   51
would                   51, 94, 148, 158, 159
wouldn                  89
written                 112
www                     6
yes                     99
you                     95, 108, 109, 110, 110, 117, 118, 119, 122, 126, 155, 161, 179, 179, 17
your                    179
```

   Note the line for "you" goes off the page. In IDLE, it will wrap around, and the last few numbers are `179, 179, 179, 181, 182`. Don't worry about the length of the lines.

   Submit. Name your file "xref2.py" and submit it to the Homework #5 area for this class on SmartSite.