COSC 311   FALL 2015   Project# 5   Hashing to build a symbol table

Distributed: 12/2/2015   **Due**: 12/14/2015

(1) Given Java source code, you will build hash tables for all legal Java names.

You should use the built-in Java support for hashing.

There will be two hash tables:

(1) A hash table for all 53 Java reserved words (50 keywords and 3 reserved words for literals). This table is named "reserve". There is a list of these words available at https://en.wikipedia.org/wiki/List_of_Java_keywords

(2) A hash table for all names in the source file that are not reserved. This table is named "name". For a definition of legal Java names, see http://docs.oracle.com/javase/tutorial/java/nutsandbolts/variables.html

The first hash table (reserved words) will store only the word itself.

The second hash table (names) will store the name and all line numbers in the source file where the name was encountered.
A reserved word cannot be used as a variable or method name. A reserved word must not be placed in the second hash table (i.e., check the reserved table before using name table)

(2) After your program has the two hash tables, it will then produce (an unsorted) cross-listing of all names in the program.

A cross-listing is an optional output of a compiler. It is a listing of names and all line numbers where the name appears.

You may scan the input file a second time in order to produce the cross-listing.

Example:
Input file – Source.java -- I have included line # for our convenience. Obviously, the line numbers do not appear in the source file.

```
1     import java.util.*;
2
3     public class Source {
4
5          public static void main (String args[]) {
6               int maxC = 50;
7               int [] arr = new int[maxC];
8
9               for (int i = 0; i < maxC; i++) {
10                  arr[i] = i * 100;
11              }
12              System.out.println("All done");
13          }
14      }
```

Will produce this output:

```
Reserved
  import
  public
  class
  void
  int
  new
  for
```

-- note, if you prefer, you can output the entire Reserved table, all 53 words.

```
Name
  java.util.              1
  Source                  3
  main                    5
  String                  5
  args                    5
  maxC                    6, 7, 9
  arr                     7, 10
  i                       9, 9, 9, 10, 10
  System.out.println      12
```

Turn in:

Hard copy of code
Hard copy of input file (can be on screen shot)
Screen shot of code running
Hard copy of contents of tables (can be on screen shot)

Run your program on the input file given above.

Grade based on:

Correct functioning
Readable code
Code design