Programming Exam: What Three Words

In this exam, you will complete the development of a Java class with two methods: one that can represent an `int` value as three words taken from a given dictionary and a second that can take any three-word representation so produced back to that same `int` value.

Your task. Add code the file `What3Words.java` that you have downloaded to create a class with a constructor that takes a file name as argument (which specifies the dictionary of words to use) and `encode()` and `decode()` methods that do the work. The test client reads integers from standard input, calls `encode()` to get the three-word representation, and calls `decode()` to check that it returns the original `int` value.

Example. Test your program with the dictionary `words5-knuth.txt` (a famous list of five-letter words compiled by D. E. Knuth) and the input file `testW3Wtiny.txt`, which has three test values. Your program must behave as follows (although it is not required that your program produce the same three-word representation as does our reference implementation).

```
% more testW3Wtiny.txt
1111111111
1234567890
2147482647

% java What3Words words5-knuth.txt < testW3Wtiny.txt
below.stows.epees  1111111111
until.beaks.clink  1234567890
later.wryer.sails  2147482647
```

Hint. If we were talking about six-digit decimal numbers and a dictionary of size 100, you might encode `123456` by using the 12th, 34th, and 56th word in the dictionary.

Requirements. As indicated by the given code in `What3Words.java`, you need to read the dictionary into an array and then build and use a symbol table to get the job done. Beyond this general approach, you must adhere to the following requirements.

- Do not add any instance variables or use any module other than those we have provided.
- If there are N different words in the dictionary, your `encode()` method must work for every nonnegative `int` value that is less than N^3 or 2147483647, whichever is smaller.
- Your `decode()` method must work for any triple produced by your `encode()` method.
- Your implementations must make reasonable use of space and time.
- For full credit, your code should raise a `java.lang.IllegalArgumentException` when given bad inputs, but do not spend much time on this aspect until you are sure that it will work for good inputs.

Note. This exam is inspired by the 2013 startup **what3words**, which implements a geocoding system allowing the use of three words to reference any of 57 trillion 3 meter by 3 meter squares on the earth. For example, Hoagie Haven is at `zoom.shark.shrimp`. You might enjoy downloading and using the app (you can easily use it to specify locations in map searches), or using your own program to help remember the values of large integers like your ID number.

SUBMIT `What3Words.java` AS INSTRUCTED ON THE COVER PAGE.