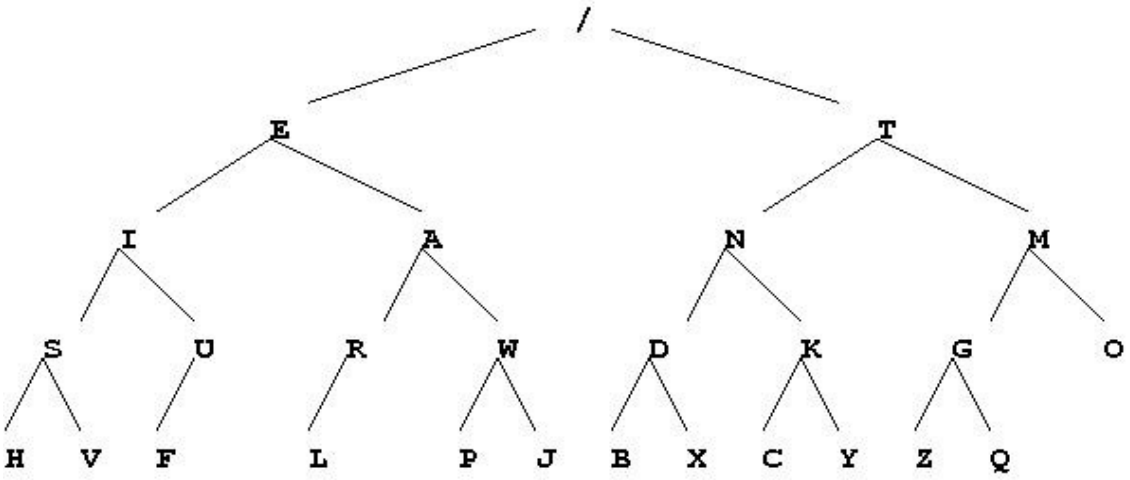
CIS 203: Assignment 10 – Decoding Morse Code

Description

Morse code is a system of transmitting text as a series of on-off clicks. The International Morse Code encodes each letter of the Roman alphabet and each Arabic numeral with a sequence of "dots" and "dashes". For example, the code for the letter 'R' is `.-.` and for 'X' it is `-..-`
In this assignment, you will only be concerned with the Morse codes for the letters of the Roman alphabet.

The Morse codes can be represented by a tree.



To get the code for a letter, follow the path to the letter. Every time you take a left, add a `.` to the code. Every time you take a right, add a `-` to the code. For example, starting at the root, we go right (`-`), then left (`.`), then left again (`.`), and then right (`-`) to get to `'X'`. Therefore the code for `'X'` is `-..-`
Here are some other codes to check your understanding of the tree:

A	.-
W	.--
V	...-
Y	-.--
E	.
T	-

You can also use the tree for getting the letter corresponding to a code. Take the code, for example, `-..` Starting at the root of the tree and at the beginning of the code take a right (`-`), a left (`.`), and another left (`.`) to get the letter `'D'`.

In this assignment you will be given a file that is encoded in Morse code. Your job is to write a program to decode it. Here is an example of an encoded file:

```
-|....|..|... ..|... .-|...|--|.-|.-..|.-..
..-.|..|.-..|. -|---|"|-|.|...|-|"|!
```

The words in the file are separated by blanks (shown above in yellow) and by line changes. The codes for the individual letters in each word are separated by pipes (`|`). The above file decoded is:

```
THIS IS A SMALL
FILE TO "TEST"!
```

Notice that characters that are not alphabetic are not coded.

You will decode a file that you are given.

What to Do

1. You have been given a class called `MorseCode` in a file called `MorseCode.java`. This class contains a code tree as described above. Write a **class** method, `getLetter(String s)`. This method takes a `String` that is Morse code for a letter and returns the `char` that it represents. For example the call:

   ```
   MorseCode.getLetter("-..")
   ```

   would return the letter `'D'`.
2. Write a class `Decode`. This class opens the file `coded.txt` that you have been given as part of this assignment, and produces an output file called `decoded.txt`. Of course, the output file is a decoded version of the `coded.txt`. You will need to read the input file line by line, decode each line and write it to the output file. Of course, to decode each line, you will need to decode each "word" on a line; and to decode each word on a line, you will need to call `MorseCode.getLetter()` to decode each letter in a word. It is up to you to design this class. Of course, you do **not** want to call `MorseCode.getLetter()` on a `String` unless it represents a Morse code.

   Suggestion: given the token

   ```
   -...|.|.-|-|?
   ```

   write a method that takes this token and returns a `String[]` or an `ArrayList<String>` that contains all the individual codes (or uncoded characters). For the above token, this array would look like this:

-...	.	.-	-	?

Now you can go through this array element by element and call `MorseCode.getLetter()` with each element that represents a legitimate Morse code. You may well want a method that you call with a `String` to determine if a `String` is a Morse code. You may make the following simplification: There are no naturally occurring periods (dots) or hyphens (dashes) in the original uncoded file.

Submission

1. Electronic submission of `MorseCode.java` and `Decode.java.`
2. Hard copy submission of the same class by the lecture time on the due date.

Grading

1. Behavior – decodes a file of Morse code that is in the format described – 45 pts
2. Comments, code style, formatting – 24 pts

Total 69 pts