Zane Durkin

Professor Bruce Bolden

Cs 121

3 February 2017

We will use four files. Each will be explained below:

**morseCodes.csv**

* This file will hold a list of key value pairs of lowercase letters and numbers and their Morse code equivalent
* Ex: [“a”,”.-“],[“b”, “-…”], …, [“1”, “.--.”], …

**headerBuilder.cpp**

* This file will read the morseCodes.csv and generate a structure and linked list using the key value pairs from the morseCodes.csv to fill in the fields of the nodes.
* Each node will contain a field for the letter/number, a field for the Morse code equivalent, and finally a field that will point to the next node in the list.
* This file will generate header.h, filling it with the a struct and all of the nodes created.

**header.h**

* This file will contain the struct and nodes that were generated form headerBuilder.cpp
* Nodes will contain a letter/number and the Morse code equivalent, as well as a pointer to the next node.

**translator.cpp**

* This file will take a text file as an input and will import the struct and nodes from header.h
* As the code transverses through each character in the text file, it will see if the character is a letter/number or if it is a dot/dash.
  + If the character is a letter/number, the code will find a node in the linked list that matches the letter/number and will return the node’s Morse code value. If the letter/number is not found in the list, the code will default to assuming that the given character does not have a Morse code value and the character will be ignored.
  + If the character is a dot/dash, the code will continue reading dots/dashes into a string until it comes across a space character. Then it will search through the linked list until it finds a node whose Morse code value matches, and then it output the corresponding letter/number from the node. If the Morse code value is not found in the list, the code’s default will be to assume that this set of dots and dashes does not have a letter/number equivalent and the character(s) will be ignored.