

Lab 5 Design Document

Methods

ZipCodeUtil

- **calculateCheckDigit:** Takes in a zip code, add up all the digits, and make the sum a multiple of 10. The number that was added to the sum to make it a multiple of 10 is our check digit, and is returned.
- **encodeDigit:** Takes in a digit. Encodes it according to the table with the weights. Outputs the bar code.
- **encodeZipCode:** Takes in a zip code. Separate it into its individual digits, call encodeDigit and return all the results as one string.
- **decodeSegment:** Takes in a bar code segment that represents one digit. Returns a digit between 0–9, otherwise, -1 is returned.
- **decodeBarCode:** Takes in a bar code. Decodes using the weight formula and returns the digit. The digit 0 is the only exception.
- **isValidBarCode:** Takes in a bar code string and checks if it is valid. The string must meet the following conditions to return true:
 - Starts and ends with full bars
 - Must only contain full bar (“|”) and half bar (“.”) characters
 - Must be exactly 32 characters long (including the starting and ending bars). Got this from figure 9
- **isValidZipCode:** Takes in a zip code. Code must meet the following criteria:
 - Must be 5 digits long

ZipCodeDemo

- **main:** Entry point to the program. Starts off by asking if the user wants to enter either (1) encode a zip code, or (2) decode a bar code. Both inputs will be taken in as a String by the Scanner, but only if they chose the former will it be cast to an integer and passed to

Public Interface

ZipCodeUtil

```
public static int calculateCheckDigit(int zipCode);
```

```
public static String encodeDigit(int digit);  
public static String encodeZipCode(int zipCode);  
public static int decodeSegment(String segment);  
public static int decodeBarCode(String barCode);  
public static boolean isValidBarCode(String barCode);  
public static boolean isValidZipCode(int zipCode);
```

ZipCodeDemo

```
public static void main();
```

Instance Variables

None needed, I think, Everything should be able to be contained within their own functions.