

TestNotes/pset2/BarCode.java

```
1 // BarCode.java
2
3 /**
4  * PSET2 #4
5  * A class representing a postal bar code and its corresponding ZIP code
6  *
7  * @author Kuljit Takhar
8  * @version October 3, 2023
9  *
10 * */
11
12 public class BarCode {
13
14     // Private instance variables representing ZIP code and barcode
15
16     private String myZipCode;
17     private String myBarCode;
18
19     /**
20      * Constructor for creating BarCode object with a ZIP code or barcode
21      * input is a 5-digit ZIP code or a 32-character barcode
22      * IllegalArgumentException is thrown if input is invalid
23      * */
24
25     public BarCode(String input) {
26         if (input.length() == 5 && isValidZipCode(input)) {
27             myZipCode = input;
28             myBarCode = encode(input);
29         } else if (input.length() == 32 && isValidBarCode(input)) {
30             myBarCode = input;
31             myZipCode = decode(input);
32         } else {
33             throw new IllegalArgumentException("Invalid input");
34         }
35     }
36
37     /**
38      * Get ZIP code associated with this bar code
39      * Return barcode as a string
40      * */
41
42     public String getZipCode() {
43         return myZipCode;
44     }
45
46     public String getBarCode() {
47         return myBarCode;
48     }
49
50     /**
51      * Converts a digit (0-9) to a 5-character bar code segment
52      * digit is the digit to convert
53      * Return the bar code segment as a string
```

```
54     */
55
56     private String digitToCode(int digit) {
57
58         // Define the encoding table
59         String[] encodingTable = {
60             "| | ::",
61             ":: | |",
62             ":: | |",
63             ":: | |",
64             ":: | |",
65             ":: | |",
66             ":: | |",
67             "| | ::",
68             "| | ::",
69             "| | ::"
70         };
71
72         return encodingTable[digit];
73     }
74
75     private int codeToDigit(String code) {
76
77         // Define the decoding table
78         String[] decodingTable = {
79             "| | ::",
80             ":: | |",
81             ":: | |",
82             ":: | |",
83             ":: | |",
84             ":: | |",
85             ":: | |",
86             "| | ::",
87             "| | ::",
88             "| | ::"
89         };
90
91         for (int i = 0; i < decodingTable.length; i++) {
92             if (code.equals(decodingTable[i])) {
93                 return i;
94             }
95         }
96
97         // Handle special case for digit 0
98         if (code.equals("| | ::")) {
99             return 0;
100         }
101
102         throw new IllegalArgumentException("Invalid barcode segment");
103     }
104
105     private boolean isValidBarCode(String barcode) {
106
107         // Check barcode length and format
108         if (barcode.length() != 32 || !barcode.matches("[|:]+")) {
109             return false;
110         }
111     }
```

```
110     }
111
112     // Check frame bars
113     if (!barcode.startsWith("|") || !barcode.endsWith("|")) {
114         return false;
115     }
116
117     // Check barcode segments
118     for (int i = 1; i <= 30; i += 5) {
119         String segment = barcode.substring(i, i + 5);
120         if (codeToDigit(segment) == -1) {
121             return false;
122         }
123     }
124
125     // Check check digit
126     String checkSegment = barcode.substring(31);
127     int checkDigit = codeToDigit(checkSegment);
128     int sum = 0;
129     for (int i = 0; i < 30; i += 5) {
130         sum += codeToDigit(barcode.substring(i, i + 5));
131     }
132     if ((sum + checkDigit) % 10 != 0) {
133         return false;
134     }
135
136     return true;
137 }
138
139 private boolean isValidZipCode(String zipCode) {
140     // Check zip code format (5 digits)
141     return zipCode.matches("\\d{5}");
142 }
143
144 private int getCheckDigit(String zipCode) {
145     int sum = 0;
146     for (int i = 0; i < 5; i++) {
147         sum += Character.getNumericValue(zipCode.charAt(i));
148     }
149     return (10 - (sum % 10)) % 10;
150 }
151
152 private String encode(String zipCode) {
153     StringBuilder barcode = new StringBuilder("|"); // Start with the frame bar
154
155     // Encode each digit and add to the barcode
156     for (int i = 0; i < 5; i++) {
157         int digit = Character.getNumericValue(zipCode.charAt(i));
158         barcode.append(digitToCode(digit));
159     }
160
161     // Add the check digit
162     int checkDigit = getCheckDigit(zipCode);
163     barcode.append(digitToCode(checkDigit));
164
165     // End with the frame bar
```

```
166         barcode.append("|");
167
168         return barcode.toString();
169     }
170
171     private String decode(String barcode) {
172         StringBuilder zipCode = new StringBuilder();
173
174         // Decode each digit segment
175         for (int i = 1; i <= 30; i += 5) {
176             String segment = barcode.substring(i, i + 5);
177             int digit = codeToDigit(segment);
178             zipCode.append(digit);
179         }
180
181         return zipCode.toString();
182     }
183 }
184
```