HOWTO: Determine if a Code 39 barcode has a checksum

In Code 39 barcodes, there is an optional convention that the last character could be a checksum of the previous characters. There is nothing encoded into the barcode that will determine if the convention is being followed -- it is up to the the writer and reader of the barcode to agree on whether the convention is being followed or not.

If you are in a situation where you have barcodes and you don't know if the last character is a checksum, you can only probabilistically determine if it is using the following procedure.

Ensure that the Barcode reader is not enforcing checksums

```
ReadOpts opts = new ReadOpts(); opts.EnforceChecksum = false;
```

Check the last character of Code 39 recognized barcodes against the previous ones to see if it is the correct checksum

This code from the WikiPedia (http://en.wikipedia.org/wiki/Code_39) calculates the Code 39 checksum. Pass in the first len-1 characters from the string and the return value should match the last character of the recognized string:

```
ublic string ValidateMod43(string barcode) { int subtotal = 0; const string charSet =
"0123456789ABCDEFGHIJKLMNOPQRSTUVWXYZ-. $/+%"; for (int i = 0; i < barcode.Length; i++) {
subtotal += charSet.IndexOf(barcode.Substring(i, 1)); } return charSet.Substring(subtotal%43,
1); }
```

If the character matches, then there is a high probability that this is a checksum character. If the character does not match, then there is a high probability that this character is not a checksum. It is not a guarantee because the characters could have been read wrong or the last character could randomly match what the checksum would be -- but if you don't know what convention is being followed, this is the best that you can do.

Original Article:

Q10237 - HOWTO: Determining if a Code 39 barcode has a checksum

Atalasoft Knowledge Base
https://www.atalasoft.com/kb2/KB/50228/HOWTO-Determine-if-a-Code-39-barcode...