

# CS1133, Spr 2018, HW 8 (Lect 17, Pts=50)

Due April 3, 2018 (before 11:00 pm)

## Problem 1 [50 pts]

Given a UPC code, correct it if one and only one of its digit is missing or undetermined (represented by an NaN), otherwise the code must be left unmodified.

You may want to work with the following given matrix:

```
[ 7  2  2  8  3  5  7  9  0  1  4  6
NaN  2  2  8  3  5  7  9  0  1  4  6
 7  2  2  8  3  5  7  9  0  1  4 NaN
 7  2  2  8  3 NaN NaN 9 NaN 1  4  6
 7  2  2  8  3  5 NaN 9  0  1  4 NaN
 2  8  3  5  7  9  0  1  4  6 NaN 8
 0  1  4  6 NaN 8  5  9  3  3  3  2
NaN 8  5  9  3  3  3  2  9  0  1  2
 1 NaN NaN 9  3  3  3  2  9 NaN 1  2
 9  2  8  3  5  7  0  1  4  6 NaN 1]
```

Use an index to pick one of the UPC code to work with. No need to write a program to process all these UPC codes. Just work with one UPC code at a time. Do not use any vectorized arithmetic, operations, or functions. Write your program to be as efficient as possible.

You may want to use a variable, to count the number of NaN encountered while reading the code from the left to the right. Continue to read and process the code as long as the total number of NaN is not greater than 1, and there are digits to be read and dealt with.

At least try the following 5 cases. The output results should be as follows:

For the UPC in row 1:

This UPC code has no bad digit.

For the UPC in row 2:

The NaN is located at position 1

Its value is computed to be 7

The corrected UPC is: 7 2 2 8 3 5 7 9 0 1 4 6

For the UPC in row 3:

The NaN is located at position 12

Its value is computed to be 6

The corrected UPC is: 7 2 2 8 3 5 7 9 0 1 4 6

For the UPC in row 4:

Too many bad digits to be corrected!

For the UPC in row 5:

Too many bad digits to be corrected!