# Bar Code Exercise

For faster sorting of letters, the United States Postal Service encourages companies that send large volumes of mail to use a bar code denoting the ZIP code. The encoding scheme for a five-digit ZIP code is shown in Table 1. There are full-height frame bars on each side. The five encoded digits are followed by a correction digit, which is computed as follows: Add up all digits, and choose the correction digit to make the sum a multiple of 10. For example, the ZIP code 95014 has sum of digits 19, so the correction digit is 1 to make the sum equal to 20.

Each digit of the ZIP code, and the correction digit, is encoded according to Table 1, where 0 denotes a half bar and 1 a full bar. Note that they represent all combinations of two full and three half bars. The digit can be easily computed from the bar code. For example, 01100 is $0 \times 7 + 1 \times 4 + 1 \times 2 + 0 \times 1 + 0 \times 0 = 6$. The only exception is 0, which would yield 11 according weight formula.

+ use at least one function
+ do not read digits separately

Table 1

```
1   0 0 0 1 1
2   0 0 1 0 1
3   0 0 1 1 0
4   0 1 0 0 1
5   0 1 0 1 0
6   0 1 1 0 0
7   1 0 0 0 1
8   1 0 0 1 0
9   1 0 1 0 0
0   1 1 0 0 0
```

**Write a program that asks the user for a ZIP code and prints the bar code.  Use : for half bars, | for full bars. For example, 95014 becomes**

```
||:|:::|:|:||:::::||:|::|:::|||
```

**Where the first and last bar represent frame bars.  The general format is:**

*Frame-bar Digit1 Digit2 Digit3 Digit4 Digit5  Check-digit Frame-bar*