Runaround Numbers

Problem Description

Runaround numbers are integers with **unique digits, none of which is zero** (e.g., 81362) that also have an interesting property, exemplified by this example:

- ☐ If you start at the left digit (8 in our number) and count that number of digits to the right (wrapping back to the first digit when no digits on the right are available), you'll end up at a new digit (**a number which does not end up at a new digit is not a Runaround Number**). The digits we run through starting from the digit 8 in the number 8 1 3 6 2 are: 1, 3, 6, 2, 8, 1, 3, and 6. So the next digit is 6.
- ☐ Repeat this process of counting for digit 6, and you should end on a new digit: 2 8 1 3 6 2, namely 2.
- ☐ Repeat again (two digits this time): 8 1
- ☐ Continue again (one digit this time): 3
- ☐ One more time: 6 2 8 and you have ended up where you started, after touching each digit once. **If you don't end up where you started after touching each digit once, your number is not a Runaround number**.

Given a number **M** (that has anywhere from 1 through 7 digits), find and print the next runaround number higher than **M**, which will not exceed 10,000,000 for the given test data.

Input

A single line with a single integer, **M**

Output

A single line containing the next runaround number **higher than** the input value, **M**

Sample Input

```
81361
```

Sample Output

```
81362
```

Program Submission

Submit your solution as a file named ***Lab03g<LabGroupNo><MatricNo>.zip*** into the right folder.