

题目描述：

程序员小明打了一辆出租车去上班。出于职业敏感，他注意到这辆出租车的计费表有点问题，总是偏大。

出租车司机解释说他不喜欢数字 4，所以改装了计费表，任何数字位置遇到数字 4 就直接跳过，其余功能都正常。

比如：

1. 23 再多一块钱就变为 25；

2. 39 再多一块钱变为 50；

3. 399 再多一块钱变为 500；

小明识破了司机的伎俩，准备利用自己的学识打败司机的阴谋。

给出计费表的表面读数，返回实际产生的费用。

输入描述：

只有一行，数字 N ，表示里程表的读数。

($1 \leq N \leq 88888888$)。

输出描述：

一个数字，表示实际产生的费用。以回车结束。

补充说明：

示例 1

输入：

5

输出：

4

说明:
<p>5 表示计费表的表面读数。</p> <p>4 表示实际产生的费用其实只有 4 块钱。</p>
示例 2
输入:
17
输出:
15
说明:
<p>17 表示计费表的表面读数。</p> <p>15 表示实际产生的费用其实只有 15 块钱。</p>
示例 3
输入:
100
输出:
81
说明:
100 表示计费表的表面读数。

81 表示实际产生的费用其实只有 81 块钱。

```
import java.util.Scanner;
```

```
// 注意类名必须为 Main, 不要有任何 package xxx 信息
```

```
public class Main {  
    public static void main(String[] args) {  
        Scanner in = new Scanner(System.in);  
        // 注意 hasNext 和 hasNextLine 的区别  
        int n = in.nextInt();  
        in.close();  
  
        int nL = 0; //n 的位数  
        while (n >= Math.pow(10, nL)) {  
            nL++;  
        }  
  
        for (int i = nL; i > 0; i--) {  
            int gw = n % (int)Math.pow(10, i) / (int)Math.pow(10, i - 1); //i 的位数  
            if (gw > 4) {  
                n -= (int)Math.pow(10, i - 1);  
            }  
            int a = n / (int)Math.pow(10, i - 1);  
            int b = a / 10;  
  
            n -= b * (int)Math.pow(10, i - 1);  
        }  
        System.out.println(n);  
    }  
}
```