Problem 3 – Saddy Kopper

In the magical land of **Telegwarts** there was a great magician, named **Saddy Kopper**. He is a nice little boy with great intentions and **Odal rune*** tattooed on his forehead. Saddy always uses his skills for good purposes like solving **mathematical tasks**. The most common show he performs is transforming numbers from one to another. Most of the time he is so fast and great that the public is unable to follow his tricks and truly believes he possesses power beyond this world.

The problem is Saddy gets easily tired. He has quite a lot of performances today that he needs your little programming skills. Saddy decided to reveal you his greatest secrets so that you can build your program in time.

The steps for performing the mathematical transformation follow:

1. Saddy gets a random positive **number** from the public
2. Saddy **removes the last digit** (the rightmost one) from the **number**
3. Saddy then finds **all digits at even positions** (starting from zero) and finds their **sum**
4. Saddy **returns again to the step 2** and **continues until no digits are left**
5. Saddy then finds the **product of all sums** found from the current **number**
6. Saddy then takes the **product** as **a new number** and in this way one **transformation** occurred
7. If the new **number** has **only one digit**, Saddy **stops** the magic trick and tells the public **how many transformations occurred in total** and what is the **final resulted number**
8. Else if this is **not the 10th transformation** in turn, Saddy gets the **new number** and **starts again with it** from **step 2**
9. If this is **the 10th transformation** in turn, Saddy **stops the magic** trick and tells the public the **final resulted number**

Example:

1. Saddy gets number **12345** from the public
2. Saddy removes the last digit – **1234** and calculates the sum of all digits at even positions – 1 + 3= **4**
3. Saddy removes the last digit – **123** and calculates the sum of all digits at even positions – 1 + 3 = **4**
4. Saddy removes the last digit – **12** and calculates the sum of all digits at even positions – **1**
5. Saddy removes the last digit – **1** and calculates the sum of all digits at even positions – **1**
6. Saddy removes the last digit – no digits left – get the product of all sums found – 4 * 4 * 1 * 1 = **16**
7. One transformation occurred and the number **16** has more than 1 digit – start again from step 2
8. Saddy removes the last digit – **1** and calculates the sum of all digits at even positions – **1**
9. Saddy removes the last digit – no digits left – get the product of all sums found – **1**
10. Second transformation occurred and the number **1** has only one digit – the magic trick stops
11. Final result – **2** transformations and the resulted number is **1**

Your task is to make a program that simulates the magic trick.

 *Check it out on Google - it will help you become better programmer ;)

Input

The input data should be read from the console.

On the only input line you will receive the number from the public.

The input data will always be valid and in the format described. There is no need to check it explicitly.

Output

The output data should be printed on the console.

If there were less than 10 transformations – print the number of transformations on the first line and the resulted number on the second line of the output.

If there were 10 transformations and the magic trick stopped – print only the resulted number on the first line of the output.

Constraints

- The **number** from the public will be between **10,000,000** and **99,999,999,999,999,999**
- Allowed working time for your program: **0.1** seconds.
- Allowed memory: **16 MB**.

Examples

Example input	Example output
123456789	3 9
2613994514047494	36100