Problem F. All Prime Day

TimeLimit:	0.5 seconds
MemoryLimit:	256 megabytes

One day, Pu Pu was browsing social media and discovered that today is an all prime day! She was so excited that she decided to post immediately on the next all prime day. Since she eagerly waits for each such day, she needs your help to write a program that finds the next all prime day after today.

Pu Pu is just a student, and her phone only supports years up to 9999. If the next day exceeds the year 9999, her phone cannot display it.

An all prime day is a day with a date in the format YYYYMMDD (an 8-digit number) such that:

- Each suffix of the number (obtained by removing zero or more digits from the left) is a prime number.

For example, the previous all prime day is 20190823, then 20190823, 0190823, 190823, 90823, 0823, 823, 23, 3 should all be prime numbers (ignoring leading zeros for suffixes).

Input

You are given a date in the format YYYYMMDD — the date of the all prime day.

- The earliest possible day is 10000101 (January 1, 1000)

- The latest possible day is 99991231 (December 31, 9999)

- It is guaranteed that the input day is a valid calendar day

A valid date follows the Gregorian calendar rules: a year is a leap year if it is divisible by 4, except for years divisible by 100, unless divisible by 400. In leap years, February has 29 days; in common years, February has 28 days. The other months have their usual number of days (January 31, March 31, April 30, etc.). You can assume the input date is always valid.

Output

Print a date in the format YYYYMMDD. If the next possible all prime day exceeds the year 9999, output "Too much digits!!!".

Examples

standard input	standard output
20251108	20300317
20190822	20190823
20190823	20300317
99990807	Too much digits!!!