CMPUT 274 - Tangible Computing **Fall 2019**
Morning Problem: Primality Testing

Description

Prime numbers are extremely important. You already know how important they are for encrypting secret messages.

An integer n is called **prime** if it is greater than 1 and only divisible by 1 and n. Given an integer, tell me if it is prime or not.

Be careful of the running time of your solution. A solution that tries to divide a number n by everything from 2 to $n - 1$ will time out.

Hint: If $n = a * b$, then we know both a and b cannot exceed $\sqrt{n}$. Neat!

Input

Input consists of a single line containing a single integer n. This number is guaranteed to be between 1 and $4,000,000,000$.

Output

Output a single line containing the text `prime` or `not prime`, indicating whether n is prime or not.

Sample Input 1

```
7
```

Sample Output 1

```
prime
```

Explanation: 7 is not divisible by any number from 2 to 6.

Sample Input 2

```
12
```

Sample Output 2

```
not prime
```

Explanation: 12 is divisible by 2, so it is not prime.

Sample Input 3

```
49
```

Sample Output 3

```
not prime
```

Explanation: 49 is divisible by 7, so it is not prime.

Sample Input 4

```
2147483647
```

Sample Output 4

```
prime
```

Explanation: 2147483647 is a prime, but you have to trust me *(or check it with your program!)*

Sample Input 5

```
1
```

Sample Output 5

```
not prime
```

Explanation: 1 is not a prime, a prime has to be greater than 1.