

# Problem E

## Enlarging Hash Tables


**Problem ID:** enlarginghashta

**CPU Time limit:** 1 second

**Memory limit:** 1024 MB

**Author:** Greg Hamerly

**Source:** Baylor Competitive Learning course

**License:** 

When implementing an open addressing hash table, we can use many different types of collision resolution strategies. One such strategy is quadratic probing. If we use quadratic probing, it's very important that the size of the hash table is a prime number. If it is a prime number and the table is less than half full, then we can prove that the quadratic probing algorithm will always be able to find an empty slot. That's necessary because finding an empty slot is the way the algorithm knows when to stop probing. However, if the table size is not prime, then even if the table is much less than half full, the probing algorithm may probe the same slots repeatedly and not find any empty slots.

Of course, even if the table size is prime, when the table gets half-full, the table size needs to increase to a larger prime. Our strategy for increasing the table size is to pick the smallest prime number larger than  $2n$ , where  $n$  was the old table size.

### Input

Input is a sequence of up to 1 000 table sizes, one per line. Each is given as an integer  $2 \leq n < 2^{30}$ . The end of input is indicated by  $n = 0$ , which should not be processed.

### Output

Output the smallest prime number larger than  $2n$ . Also, if  $n$  happens not to be prime, add the parenthetical note '( $n$  is not prime)'.

#### Sample Input 1

```
29
33
0
```

#### Sample Output 1

```
59
67 (33 is not prime)
```