Solovay-Strassen Inverted

Let us denote the set of all prime numbers by the symbol **P**. The Solovay-Strassen algorithm determines whether a given positive odd integer $n>2$ belongs to **P**.

The *Legendre function* sig for number $n \in$ **N** with parameter $s \in$ **N** ($s<n$) is defined by the formula $\text{sig}(n,s)=s^{(n-1)/2} \bmod n$. The symbol mod is defined in such a way as to return the result with the smallest possible absolute value, from the range (-n/2, n/2].

The *Jacobi function* jac for number $n \in$ **N** with parameter $s \in$ **N** ($s<n$) is given as:

$$\text{jac}(n,s)=\begin{cases} \text{sig}(n,s), & \text{if } n \in \mathbf{P} \\ \displaystyle\prod_{i=1}^{k}\text{sig}(p_i,s), & \text{if } n=\displaystyle\prod_{i=1}^{k}p_i, \text{ where all } p_i \in \mathbf{P} \end{cases}$$

It is interesting to note that for given n and s, the values of sig(n,s) and jac(n,s) can be computed in $O((\log_2 n)^2)$ time. For particulars consult an encyclopedia, such as MathWorld.

The deterministic version of the Solovay-Strassen primality-test algorithm is given below.

```
algorithm Solovay-Strassen (n)
var s;
begin
   for s in {1,2,3,4,...,n} do
      if sig(n,s)≠jac(n,s)
         then return "n is composite (detected at attempt <s>)";
   return "n is prime";
end.
```

Task

We are not asking you to implement the Solovay-Strassen algorithm, this would be far too easy :). Instead, try to find values of n, for which the output of the algorithm would be "n is composite (detected at attempt 1)", "n is composite (detected at attempt 2)", and so on. Write out as many of these values as you can in consecutive lines, and try to keep them as small as possible.

Scoring

The score awarded to your program is the total of all points given for its individual lines.

The i-th line output by your program should contain exactly one positive odd integer $n>2$ of no more than 500 decimal digits. If Solovay-Strassen(n) yields the answer "n is composite (detected at attempt i)", you will receive $i/\log_{10} n$ points for this line, if not - your program will be considered incorrect. Output 0 if you don't want a line to be assessed. Only the first 1000 lines of output are taken into account.

Example

A program outputing:

```
0
0
561
```

will receive $3/\log_{10} 561 = 1.091$ points.