Converting Leutonian Numbers to Decimal

Instead of binary or decimal, the Kingdom of Leutonia uses an unusual system to represent numbers, based on the Fibonacci sequence. The Fibonacci sequence $F_0, F_1, F_2, \ldots$ is defined recursively as follows.

$$
\begin{aligned}
F_0 &= 1 \\
F_1 &= 1 \\
F_n &= F_{n-1} + F_{n-2} \text{ for } n \geq 2
\end{aligned}
$$

A Leutonian number is a string of 0's and 1's that begins with a 1 and never has two consecutive 1's. If $s = s_\ell s_{\ell-1} \ldots s_1$ is such a string of length ℓ, where each s_i is in $\{0, 1\}$, the number represented by s is is $\sum_{i=1}^{\ell} s_i \cdot F_i$.

For example, the number represented by 1000101 is $F_7 + F_3 + F_1 = 21 + 3 + 1 = 25$.

Input

The input sequence will be a list of Leutonian numbers, one per line. Each Leutonian number will be at most 25 characters long. The last line (which should not be processed) will contain a single 0 instead of a Leutonian number.

Output

For each Leutonian number given in the input, output the standard decimal representation of the number. Each output should appear on a separate line.

Sample Input

```
10
1000101
0
```

Sample Output

```
2
25
```