Problem B: Digit Solitaire

Source file: `digits.{c, cpp, java}`
Input file: `digits.in`

Despite the glorious fall colors in the midwest, there is a great deal of time to spend while on a train from St. Louis to Chicago. On a recent trip, we passed some time with the following game.

We start with a positive integer *S*. So long as it has more than one digit, we compute the product of its digits and repeat. For example, if starting with 95, we compute $9 \times 5 = 45$. Since 45 has more than one digit, we compute $4 \times 5 = 20$. Continuing with 20, we compute $2 \times 0 = 0$. Having reached 0, which is a single-digit number, the game is over.

As a second example, if we begin with 396, we get the following computations:
$3 \times 9 \times 6 = 162$
$1 \times 6 \times 2 = 12$
$1 \times 2 = 2$
and we stop the game having reached 2.

Input: Each line contains a single integer $1 \leq S \leq 100000$, designating the starting value. The value *S* will not have any leading zeros. A value of 0 designates the end of the input.

Output: For each nonzero input value, a single line of output should express the ordered sequence of values that are considered during the game, starting with the original value.

Example input:	**Example output:**
95 396 28 4 40 0	95 45 20 0 396 162 12 2 28 16 6 4 40 0