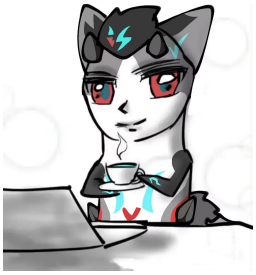
Python Will be Faster than C++

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 second
Memory limit:	256 megabytes



Little Q is learning programming languages. He often uses Python as it is one of the most used programming languages. However, when he codes in Python, he finds that the programs do not run very efficiently, especially when compared to C/C++. He wonders if the running efficiency of Python has any improvement with version updates, and conducts an experiment.

The algorithm used in the experiment is Monte Carlo, estimating π by generating a large number of random points in a square. He codes this algorithm in Python and runs it with different versions of Python, logging the running time. Formally, there are a total of n released versions of Python, namely 3.1, 3.2, ..., 3.n. The running time of the algorithm on version 3.i is a_i ms. Little Q is surprised to find that the running efficiency of Python does change with version iterations.

Little Q also tests the running time of this algorithm in C++, which is stably k ms. Then he comes up with a funny idea: using the experimental data to predict which future version of Python will have a higher efficiency than C++. Unfortunately, he forgets how to apply the regression model, so he uses a brute prediction method:

- For a future version i that $i > n$, $a_i = \max(0, 2a_{i-1} - a_{i-2})$.

With this prediction method, please tell him the earliest version of Python that has a strictly higher efficiency than C++, i.e., find the minimum i such that $a_i < k$.

Input

The first line contains two integers n and k ($2 \le n \le 10$, $1 \le k \le 1000$), indicating the number of released versions of Python and the running time of the algorithm in C++.

The second line contains n integers $a_1, a_2, \ldots, a_n$ ($k < a_i \le 10^5$), indicating the running time of the algorithm on each version of Python.

Output

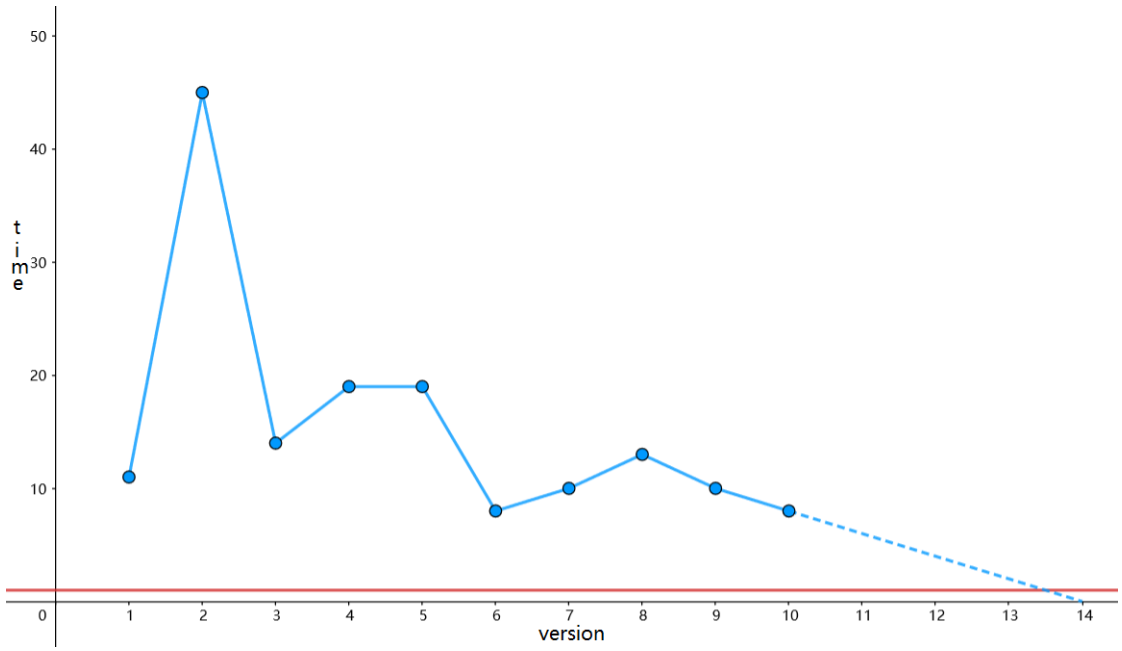
Output `Python 3.`x `will be faster than C++`, where x is an integer indicating the earliest version that $a_x < k$. If such a version does not exist, output `Python will never be faster than C++`.

Examples

standard input	standard output
10 1 11 45 14 19 19 8 10 13 10 8	Python 3.14 will be faster than C++
10 1 2 2 2 2 2 2 2 2 2 2	Python will never be faster than C++

Note

The first sample can be expressed in the following picture, in which the blue line represents the efficiency of Python, and the red line represents the efficiency of C++. The dashed line indicates the prediction. As you see, Python 3.14 will be faster than C++.



For the second sample, it is easy to notice that Python always runs for 2 ms, including the prediction of future versions. Therefore, Python will never be faster than C++, which runs for 1 ms.