# Monetary System

| | |
|---|---|
| Input file: | `standard input` |
| Output file: | `standard output` |
| Time limit: | 1 second |
| Memory limit: | 512 megabytes |

Given a sorted ascending array $A$ of length $n$, where the $i$-th element is $A_i$ with $A_1 = 1$.

Construct a currency system using this array. For any positive integer $x$, define the currency note count function as $f(x, n)$, where $n$ represents the array's length. This function represents the number of banknotes required to pay $x$ yuan under this currency system, following the greedy principle of always using the largest possible denominations first before smaller ones. For any positive integer $x$ and any positive integer $y \in [1, n]$, $f(x, y)$ satisfies:

$$f(x,y) = \begin{cases} \lfloor \frac{x}{A_y} \rfloor + f(x \bmod A_y, y-1) & y > 1 \\ x & y = 1 \end{cases}$$

You need to process $q$ queries. For each query, given an integer $m$, determine how many positive integers $x$ satisfy $f(x, n) = m$.

## Input

The first line contains two integers $n$ and $q$ ($1 \le n \le 10^5, 1 \le q \le 10^6$), representing the length of the array $A$ and the number of queries.

The second line contains $n$ positive integers $A_1, A_2, \ldots, A_n$ ($1 = A_1 < A_2 < \ldots < A_n \le 10^6$), the elements of array $A$.

The third line contains $q$ integers $m_1, m_2, \ldots, m_q$ ($1 \le m_i \le 10^9$), where $m_i$ is the value for the $i$-th query.

## Output

Output a line containing $q$ space-separated integers. The $i$-th integer is the answer to the $i$-th query.

## Example

| standard input | standard output |
|---|---|
| 6 2<br>1 5 10 20 50 100<br>1 2 | 6 18 |

## Note

For the sample, six denominations of notes are given in the monetary system: 1-yuan, 5-yuan, 10-yuan, 20-yuan, 50-yuan, and 100-yuan. When making a 6-yuan payment using notes, you must use two notes: one 1-yuan note and one 5-yuan note, i.e., $f(6, 6) = 2$. Although you could theoretically use six 1-yuan notes to make a 6-yuan payment, this method does not satisfy the principle of taking as many large denomination notes as possible and therefore does not satisfy the definition of the function in this problem.