Problem J. Johnny's Birthday

Input file:	`standard input`
Output file:	`standard output`
Time limit:	7 seconds
Memory limit:	256 mebibytes

Johnny is making a birthday party soon. Instead of candies this time he wants to treat his classmates with chocolate. He was collecting chocolate bars for a long time, even though it was very hard for him to restrain himself from eating them all! He couldn't help but to have a bite from each of the bars, though – depending on tastiness of a particular chocolate bar, his strong will, phase of the moon, and other factors he ate one or more slices of chocolate... In the end he has chocolate bars of many different shapes. But there are also many children in his class, and he needs to share the chocolate fairly.

He decided that in order to share the chocolate fairly he will perform the following operation called *bisplitting*: Johnny takes the largest of chocolate bars (in terms of the number of chocolate cubes in it) and cuts it twice in half, horizontally and vertically. Johnny doesn't want to cut chocolate cubes in half so he sometimes has to cut unevenly, but he'll always do it as close to the half as possible – if a chocolate bar had size $w \times h$, then after bisplitting he'll get bars of sizes $\lfloor \frac{w}{2} \rfloor \times \lfloor \frac{h}{2} \rfloor$, $\lfloor \frac{w}{2} \rfloor \times \lceil \frac{h}{2} \rceil$, $\lceil \frac{w}{2} \rceil \times \lfloor \frac{h}{2} \rfloor$, $\lceil \frac{w}{2} \rceil \times \lceil \frac{h}{2} \rceil$. It may happen that some of the bars he obtains in this way have one or both dimensions of length 0, that is the chocolate bar will be empty. If there's more than one chocolate bar that has maximum number of chocolate cubes, Johnny will pick from them the one with the maximum length of the longer edge.

Johnny decided that it will be best if the resulting chocolate bars won't be too big, then he'll get to keep more chocolate for himself. Thus he wants to know (and you'll have to help him with that) how many chocolate cubes will be in the largest chocolate bar after he performs k_i bisplitting operations, for $i = 1, 2, \ldots, q$ — based on that information he'll decide how long he should keep splitting the chocolate bars.

Input

The first line of input contains two integers n and q ($1 \le n \le 50\,000$, $1 \le q \le 100$), separated by a single space. They denote the number of chocolate bars Johnny has, and the number of queries Johnny will make.

Each of the next n lines contains two integers separated by a single space. i-th of those lines contains w_i and h_i, ($1 \le w_i, h_i \le 10^9$), denoting a chocolate bar with dimensions of $w_i \times h_i$ chocolate cubes.

Each of the next q lines contains a single non-negative integer. In i-th of those lines there is k_i ($0 \le k_i \le 10^{18}$). This is the query about the number of chocolate cubes that the largest chocolate bar will have after Johnny performs bisplitting operation k_i times starting from the initial set of chocolate bars.

Output

The output should contain exactly q lines. Each of those lines should contain a single integer – the answer to query in the same order as in the input.

Example

standard input	standard output
2 4 5 10 7 7 2 3 10 1	16 15 6 49

Note

The sets of chocolate bars after the first $i = 0, 1, 2, 3$ bisplittings, sorted from largest to smallest bar (chocolate bars of sizes $a \times b$ and $b \times a$ are the same and are denoted as $\min(a, b) \times \max(a, b)$):

- **Starting set:** one bar 5×10 and one bar 7×7;
- **After first bisplitting:** one bar 7×7, two bars 3×5, and two bars 2×5;
- **After second bisplitting:** one bar 4×4, two bars 3×5, two bars 3×4, two bars 2×5, and one bar 3×3;
- **After third bisplitting:** two bars 3×5, two bars 3×4, two bars 2×5, one bar 3×3, and four bars 2×2.