

Mask Allocation

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

Nowadays, the Kingdom of Dreamgrid is suffering from a national pandemic. Fortunately, president Baobao is working effectively with the Center for Disease Control (CDC) and they are trying their best to make everything under control.

President Baobao has received $n \times m$ medical masks from his friend Reku, an extremely rich billionaire. As the chief of the CDC, you are required to allocate these masks properly. There are 2 kinds of hospitals in the Kingdom of Dreamgrid, n senior hospitals for critically ill patients and m mobile cabin hospitals for patients with mild symptoms.

Before allocating them to hospitals, you have to pack them into boxes. Please note that boxes must not be opened in order to prevent contamination and you only know these boxes will be allocated to either senior hospitals or mobile cabin hospitals. That is to say, there should be a way of dividing masks boxes into m groups of n masks, and a way of dividing into n groups of m masks.

You want the number of boxes to be minimal and please also provide a lexicographically greatest sequence of the numbers of masks in boxes. We say a sequence a is lexicographically greater than another b of the same length if there exists an integer i , such that

- $a_j = b_j$, for all $j < i$; and
- $a_i > b_i$.

Input

There are multiple test cases. The first line of the input contains an integer T ($1 \leq T \leq 100$), indicating the number of test cases.

For each test case, the only line of the input contains two integers n, m ($1 \leq n, m \leq 10^4$), representing the numbers of senior hospitals and mobile cabin hospitals.

Output

For each test case, please output two lines. The first line should contain a single integer k in the first line, indicating the minimum number of boxes. In the second line, please output k integers, denoting the lexicographically greatest sequence.

Example

standard input	standard output
2	8
5 4	4 4 4 4 1 1 1 1
3 3	3
	3 3 3