



Problem A Text Formatting

Your chief judge is getting long in the tooth and it is getting increasingly difficult for him to read through the densely-written messages of complaints about the problems and scores. The chief judge prefers to have more and uniform spacing between the words, and you have been drafted to write a program to format lines of text accordingly.

Your task is to write a program to read a number of lines of text and format each line independently such that:

1. successive words on a formatted line are separated by *exactly* two blank spaces, and
2. words are NOT split between lines, and
3. width of the formatted text does not exceed a specified value.

Leading and trailing blank spaces on each given line should be ignored.

As an example:

Your chief judge is geting long in the tooth. Have a nice day.
Your chief judge is geting long in the tooth. Have a nice day.
01234567890123456789

The first input line is formatted into three (3) lines of width bounded by 20 places, and words separated by two blank spaces in each line. The second input line is formatted independently on the fourth output line.



INPUT:

Input to this problem starts with an integer K , $K > 0$, that represents the number of messages. The number K is given on a separate line followed by a description of the K messages. The description of each message starts with a line that contains two integers. The first integer W , $W \geq 20$, represents the desired width of the formatted text, and the second integer N , $N > 0$, represents the number of lines in the message. A single blank space separates the two integers. The message, which consists of N lines, follows with each line consisting of a sequence of one or more words separated by blank spaces. The length of each word is less than or equal to W . That is, a word like “supercalifragilistic-expialidocious” is only to be expected as part of the input if $W \geq 35$.

OUTPUT:

For each message the output consists of one line. The line starts with the message number (the first message being “Message 1”), followed by a “: ” (colon followed by space), as shown in the EXAMPLE OUTPUT below. This is followed by the number of lines that the formatted text would occupy.

EXAMPLE INPUT:

```
2
20 1
Your chief judge is geting long in the tooth.
30 2
For each message the output consists of one line.
The chief judge now prefers to have more and uniform spacing
```

EXAMPLE OUTPUT:

```
Message 1: 3
Message 2: 6
```