Problem D. Music Your Way

Time limit	1000 ms
Mem limit	1048576 kB
OS	Linux

MyTunes, a new music application from Mapple, allows you to display and sort your music according to many different attributes. Attributes like album name, song title, artist, and song length are available, among others. One feature MyTunes does NOT have is stable sorting. You need to write the software which does this. Stable sorting would allow the user to do something like this: first sort the whole list by song title, and then sort the whole list by artist. The result is that the list of songs are sorted primarily by artist (since it's the last thing used to sort), then within each artist they are sorted by song name (since before the artist sort, the songs were ordered by song name). When sorting two songs with equal attribute values they should remain in their relative order prior to sorting (giving stable sorting).

The first sort starts from the order of songs as given in the input. Each subsequent sort starts from the order given by the previous sort, rather than starting from the initial order. All sorting should be done least to greatest, in lexicographic order (using ASCII values, even for numeric data).

Input

The first line contains a list of 1 to 20 unique attributes, separated by single spaces. The next line contains an integer $1 \leq m \leq 100$ giving the number of songs that follow, one song per line. Each song has the values for all the attributes, separated by single spaces. The next line contains an integer $1 \leq n \leq 20$ giving the number of sorting commands. The following n lines each contain one sort command, which is the name of an attribute. Initially, the song list is in the order given on the input. All attributes and values are between 1 and 30 characters, using only a−z, A−Z, 0−9, and underscore (_).

Output

After each sort command, print the list of attributes followed by the song list in the order given by the sort command, one song per line. Print a space between adjacent fields. Between each pair of printed lists, print a blank line.

Sample 1

Input	Output
Artist Album Song_Title Length_seconds 5 Tchaikovsky Piano_Favorites November 163 James_Taylor Dad_Loves_His_Work Hard_Times 192 James_Taylor Dad_Loves_His_Work London_Town 236 James_Taylor Dad_Loves_His_Work Summers_Here 163 George_Winston Summer Lullaby 215 3 Artist Album Song_Title	Artist Album Song_Title Length_seconds George_Winston Summer Lullaby 215 James_Taylor Dad_Loves_His_Work Hard_Times 192 James_Taylor Dad_Loves_His_Work London_Town 236 James_Taylor Dad_Loves_His_Work Summers_Here 163 Tchaikovsky Piano_Favorites November 163 Artist Album Song_Title Length_seconds James_Taylor Dad_Loves_His_Work Hard_Times 192 James_Taylor Dad_Loves_His_Work London_Town 236 James_Taylor Dad_Loves_His_Work Summers_Here 163 Tchaikovsky Piano_Favorites November 163 George_Winston Summer Lullaby 215 Artist Album Song_Title Length_seconds James_Taylor Dad_Loves_His_Work Hard_Times 192 James_Taylor Dad_Loves_His_Work London_Town 236 George_Winston Summer Lullaby 215 Tchaikovsky Piano_Favorites November 163 James_Taylor Dad_Loves_His_Work Summers_Here 163

Sample 2

Input	Output
Artist_Name Song 2 Chopin Scherzo_no_1_in_B_minor Chopin Polonaise_no_3_in_A 2 Artist_Name Song	Artist_Name Song Chopin Scherzo_no_1_in_B_minor Chopin Polonaise_no_3_in_A Artist_Name Song Chopin Polonaise_no_3_in_A Chopin Scherzo_no_1_in_B_minor