Problem K – Keylogger

Lately you have been very curious about your typing speed, and you have been wondering how long it takes for you to press each key on your keyboard, which has K keys.

To figure that out, you installed a keylogger on your own computer. It has been registering the delta time between each pair of key presses. After collecting data for a couple of weeks, you now have access to a 2-dimensional matrix T with K rows and K columns. The element at the i-th row and j-th column is $T_{i,j}$, and it represents how long it takes, on average, for you to press the key j right after having pressed the key i. For example, the element $T_{3,5}$ represents how long it takes, on average, for you to press the key 5 right after having pressed the key 3. Coincidentally, each row on T is ordered non-decreasingly.

Given that your typing speed varies according to the time of the day and your mood, your keylogger has also given you a latency margin error L. That means that, for every pair of keys i and j on your keyboard, it actually takes between $T_{i,j} - L$ and $T_{i,j} + L$, inclusive, for you to press the key j right after having pressed the key i.

You classified for the South American ICPC regional competition, and you have been asked to update some of your contact information on the ICPC website. The problem is that you have been studying so hard that you forgot your password. All you remember is that your password has length N.

Luckily, your keylogger also has data about the last time you typed your password on that website. So now you have an array P with $N - 1$ elements. Each element P_i represents the delta time between each consecutive key presses from your password. In other words, P_1 represents the delta time between you pressing the keys of the first and the second characters of your password, P_2 is the delta time between you pressing the keys of the second and third characters of your password, and so on. Notice that the latency L does not apply to P, because each P_i is not an average but a single delta time, measured precisely.

You need to recover your password as soon as possible. Your plan now is to try every sequence of keys that is compatible with the information you have. A sequence S of length N is compatible with L, T, and P if each pair of consecutive keys S_i and S_{i+1} satisfy that $T_{S_i,S_{i+1}} - L \leq P_i \leq T_{S_i,S_{i+1}} + L$. How many such sequences are there?

Input

The first line contains two integers K ($1 \leq K \leq 750$) and L ($0 \leq L \leq 10^9$), indicating respectively how many keys are there on your keyboard and the latency margin error given by your keylogger. The next K lines contain K integers each, representing the matrix T. The j-th integer on the i-th line is $T_{i,j}$ ($1 \leq T_{i,j} \leq 10^9$ for $i = 1, 2, \ldots, K$ and $j = 1, 2, \ldots, K$). Recall that $T_{i,j}$ indicates how long it takes, on average, for you to press the key j right after having pressed the key i, and that each row on T is ordered non-decreasingly ($T_{i,j} \leq T_{i,j+1}$ for $i = 1, 2, \ldots, K$ and $j = 1, 2, \ldots, K - 1$). The next line contains an integer N ($2 \leq N \leq 10^4$), representing the length of your password. The final line contains $N-1$ integers $P_1, P_2, \ldots, P_{N-1}$ ($1 \leq P_i \leq 10^9$ for $i = 1, 2, \ldots, N - 1$), denoting the delta time between each consecutive key presses from your password.

Output

Output a single line with an integer indicating how many different sequences of keys are compatible with the information you have. Because this number can be very large, output the remainder of dividing it by $10^9 + 7$.

Sample input 1	Sample output 1
4 0 1 1 3 5 2 4 4 10 1 1 1 8 5 6 7 8 5 2 3 8 5	1

Sample input 2	Sample output 2
3 3 9 10 15 9 13 16 3 5 6 3 10 5	0

Sample input 3	Sample output 3
5 1 1 5 6 8 10 1 2 4 5 5 5 5 5 6 8 3 3 3 4 5 1 1 3 4 5 4 1 3 7	4