Problem C. Classement Nationale

Input file:	`standard input`
Output file:	`standard output`
Time limit:	3 seconds
Memory limit:	256 mebibytes

Orienteering is a sport where athletes race through forests using a map to visit control points. After arriving in France, Kostya wanted to understand the local rating system called "Classement Nationale" (CN). Here is a moderately simplified model of this rating system.

In our model of the French orienteering system, there are the following elements:

- A total of n athletes, numbered 1 through n. Kostya is athlete number 1.
- A list of m competitions (races) in the order of their occurrence. Competition j occurs on day d_j, and has a flag $s_j \in \{0, 1\}$ indicating "special" status (for example, French Championships).
- A list of k_j participants who took part in competition j. The i-th participant is athlete number a_i who finished the race with time (result) $R_{a_i,j}$. Some athletes may be disqualified (DSQ): they appear in the list with $R_{a_i,j} = 0$. Except for the DSQ case, the less the $R_{a_i,j}$ value is, the better the performance of an athlete.
- Each athlete i has a base rating $CN_{i,\text{base}}$ (their rating before any competition they participated in).
- Each day is attributed to a year, each year being 365 days long. So, days $[0; 365)$ form the first year, days $[365; 730)$ form the second year, and so on.
- During each year, there exists a normalization coefficient K_{norm}. During the first year, $K_{\text{norm}} = 10\,000$.

At each moment of time t, athlete i has a defined rating $CN_{i,t}$. Ratings evolve over time according to the following process.

1. For each competition on day d_j:
   (a) Fetch each participant's rating from 15 days ago: $CN_{i,(d_j-15)}$.
   (b) Calculate $F_j = |\{i : R_{i,j} \neq 0\}|$, the number of participants who finished the race without being disqualified.
   (c) Compute the *RaceScore*:
   $$N_{\text{infl}} = \max\big(3, \lfloor 2F_j/3 \rfloor\big),$$
   $$\text{and if } F_j \geq N_{\text{infl}}, \text{ then } RaceScore_j = \left\lfloor \frac{1}{N_{\text{infl}}} \sum_{p=1}^{N_{\text{infl}}} \left(R_{Pos_p,j} \cdot CN_{Pos_p,(d_j-15)}\right) \cdot \frac{10\,000}{K_{\text{norm}}} \right\rfloor$$
   where $Pos_1, \ldots, Pos_{F_j}$ lists non-disqualified participants in increasing order of $R_{i,j}$ (ties are broken in a way that an athlete with a higher CN value is earlier in rankings). If $F_j < N_{\text{infl}}$, the race is unrated and no performances are recorded.
   (d) In case the race is rated, each of the k_j athletes who took part gets a record of participation. Athlete a_i has the following performance value:
   $$Performance_{a_i,j} = \begin{cases} -1 & \text{if } R_{a_i,j} = 0 \text{ (DSQ)}, \\ \left\lfloor \frac{RaceScore_j}{R_{a_i,j}} \right\rfloor & \text{otherwise.} \end{cases}$$
   This performance (along with competition date d_j and special flag s_j) is added to athlete i's history.

2. At any day t, an athlete i's current $CN_{i,t}$ is defined by their recorded performances. Let H_i be the multiset of all performances in days $(t-365, t]$.

   (a) If $H_i = \varnothing$, then:

   $$CN_{i,t} = \begin{cases} CN_{i,\text{base}} & \text{if athlete } i \text{ has no recorded performances for all times in } (-\infty, t], \\ 0 & \text{otherwise.} \end{cases}$$

   (b) Otherwise:

   - Sort H_i in non-increasing order.
   - Let $h = \min(\max(\lfloor 0.4 \cdot |H_i| \rfloor, 4), |H_i|)$. Then:
     i. Take the h largest **non-negative** scores into multiset S. In case of ties performance in special race ($s_j = 1$) should be taken first.
     ii. Construct a multiset $S' = S$. If $|S| > 1$, drop the very best score from S' unless there exists a special race ($s_j = 1$) with the very best score. If $|S| \le 1$, S' is not changed under any circumstances.
     iii. Compute avg: the average of the scores in S'. In case $|S'| = 0$, let $avg = 0$.
     iv. If $|S| < 4$, there will be a penalty of 2% per missing race (for example, 3 races $\to$ 2%, 2 races $\to$ 4%, 1 race $\to$ 6%). If there were DSQ races in H_i, this penalty should be reduced by 2% regardless of the number of DSQ races. We define the effect of x% penalty on number y as $\frac{100-x}{100} \cdot y$.
   - The floored avg after applying the penalty is $CN_{i,t}$. Please note that this value is an integer, and was floored from the corresponding rational number exactly once in the end.

3. Additionally, whenever $t \equiv 0 \pmod{365}$ for all years but the first, a recalibration is performed (before any of the races on that day, if any):
   - Let $M = \max_i CN_{i,t-1}$.
   - If $M = 0$, $K_{\text{norm}} = 10\,000$.
   - Otherwise, $K_{\text{norm}} = M$.

Orienteering is not the most popular sport, so athletes want as many people coming to the competitions as possible. Kostya has a hunch that this complicated system has the opposite effect.

So Kostya wants to show the French Orienteering Federation that by skipping some races he actually attended, he could end up with different ratings after the last competition of the season. In other words, he is interested in different possible values of CN_{1,d_m} after all updates have been processed if he was not listed in the results of some races. As orienteering is an individual sport, all $R_{i,j}$ do not influence each other.

Given the full data of n athletes, m competitions, and Kostya's personal attendance list, compute every distinct final rating CN_{1,d_m} achievable by choosing a subset of the races Kostya attended (including an empty subset). Output the ratings in strictly decreasing order.

Input

The first line contains two integers n and m ($1 \le n, m \le 1000$): the number of athletes and the number of competitions.

The second line contains n integers $CN_{i,\text{base}}$ ($1 \le CN_{i,\text{base}} \le 12\,000$): base ratings for the athletes.

The third line contains m integers $d_1, \ldots, d_m$ ($1 \le d_i \le 2000$, $d_i \le d_{i+1}$): competition days for the races.

The fourth line contains m integers $s_1, \ldots, s_m$ ($0 \le s_i \le 1$): special flags for the races.

Then follow n blocks, one per athlete $i = 1, \ldots, n$:

- A line with an integer n_i ($0 \le n_i \le 10$), the number of races that athlete i participated in.

- Each of the next n_i lines contains two integers b_j and R_{i,b_j} ($1 \le b_j \le m$, $600 \le R_{i,b_j} \le 9000$ or $R_{i,b_j} = 0$): competition number and result in seconds. For each player, all b_j are distinct.

Output

On the first line, print a single integer c: the number of distinct achievable ratings. On the next c lines, print the achievable values of CN_{1,d_m} in decreasing order, one per line.

Examples

standard input	*standard output*
3 4 6200 7800 10200 90 100 300 464 1 1 1 1 4 1 1950 2 1940 3 900 4 1800 4 1 1920 2 1800 3 844 4 1800 4 1 1860 2 1670 3 769 4 1800	15 9156 8120 8016 7981 7920 7828 7719 7433 6983 6965 6960 6899 6600 6200 0
6 2 10118 10485 10427 9158 9307 10044 356 365 0 0 2 1 1105 2 791 1 2 960 1 1 929 1 1 841 0 2 1 1043 2 1195	4 11360 10118 8068 7900

Note

We will explain two of the achievable CN values for the first example. Skipping all the 4 races would result in $CN_{1,d_m} = CN_{1,\text{base}} = 6200$ for Kostya. If Kostya attends only the first race on day 90, by the day 464 it would be gone from the performances being considered, thus his $CN_{1,d_m} = 0$. Other possible results are given in the output.