

FAST FOOD PRIZES

Around regional contest time, the Canadian branch of a popular fast food restaurant usually runs a game to promote its business. Certain food items provide stickers, and certain collection of different stickers can be converted to cash prizes. If a prize requires sticker types T_1, T_2, \dots, T_k , then you can claim the prize if you have 1 sticker of each type T_1, T_2, \dots, T_k . Each sticker can only be used to claim one prize. However, you may claim a prize multiple times if you have multiple stickers of the same type. No two prizes will require the same type of stickers. There may be some stickers that cannot be used to claim a cash prize (e.g. a sticker for a free milkshake).

On your road trip to the regional contest, your coach forced you to eat at this restaurant and collected all the stickers together. How much cash can your coach claim?

Input

The input consists of multiple test cases. The first line of input is a single integer, not more than 1000, indicating the number of test cases to follow. Each case starts with a line containing two integers n ($1 \leq n \leq 10$) and m ($1 \leq m \leq 30$), where n is the number of different types of prizes, and m is the number of different types of stickers (the types are labelled $1, 2, \dots, m$). The next n lines specify the prizes. Each of these lines starts with an integer k ($1 \leq k \leq m$) specifying the number of sticker types required to claim the prize. This is followed by k integers specifying the types of the stickers required. The final integer on each line is the (positive) cash value of the prize (at most 1,000,000). The last line of each case gives m nonnegative integers, with the i th integer giving the number of stickers of type i your coach has collected. There are no more than 100 stickers of each type.

Output

For each case, display on a single line the total value of the cash prizes that can be claimed.

Examples

Nº	stdin	stdout
1	3	500
	2 10	2500
	3 1 2 3 100	1900
	4 4 5 6 7 200	
	2 3 1 4 5 2 2 1 3 4	
	3 6	
	2 1 2 100	
	3 3 4 5 200	

1 6 300

1 2 3 4 5 6

3 6

3 1 2 100

3 3 4 5 200

1 6 300

1 2 0 4 5 6