

Problem C. Mr. Liang play Card Game

Input file: **standard input**
Output file: **standard output**
Time limit: 3 seconds
Memory limit: 128 megabytes

Recently, Mr. Liang has become obsessed with a card game and cannot break free from it. The game goes like this: there are n cards arranged in a row from left to right. Each card has a type and a level (initially, all card levels are 1). You can perform the following operations an unlimited number of times:

Operation 1: Select a card and play it. Each card type has a value V_i . Playing a level 1 card yields a profit of V_i , playing a level 2 card yields a profit of $P \cdot V_i$, playing a level 3 card yields a profit of $P \cdot P \cdot V_i$ and so on. However, there is a restriction on the card level, with the maximum level being R .

Operation 2: Select two adjacent cards of the same type and level, and merge them into a higher-level card.

As his good friend, cv4456 would like to ask you what is the maximum profit Mr. Liang can obtain in the end?

Input

The input consists of multiple test cases. The first line contains a single integer t ($1 \leq t \leq 50$) — the number of test cases. Description of the test cases follows.

The first line of each case are four integers, n, m, R, P ($1 \leq n \leq 100, 1 \leq m \leq 20, 1 \leq R \leq 20, 1 \leq P \leq 10$), denoting the number of cards, types of cards, the upper limit of card levels, and the doubling coefficient for higher-level cards.

The second line of each case are n integers a_i ($1 \leq a_i \leq m$), denoting the types of the n cards initially placed on the table. (all cards on the table is level 1)

The third line of each case are m integers V_i ($1 \leq V_i \leq 10^5$), denoting the weight of each kinds of card.

The data guarantees that there will be no more than 10 groups with a value of n exceeding 20.

Output

For each test case print a single integer — the maximum profit Mr. Liang can obtain in the end.

Example

standard input	standard output
1 7 3 4 3 1 3 2 3 2 3 3 1 2 3	32