**Mentoring Assignment**

**The Problem**

The Adolescent Shelter is a charitable institution which runs a program to support troubled adolescents via the help of mature volunteers. The program aims to pair an adolescent with a volunteer to provide the adolescent with the best possible mentoring and to provide the volunteer with a rewarding experience. Towards that goal, each volunteer and each adolescent is required to perform a personality-profiling test. A test reports a score for each of the ***N*** different personality traits. The scores can be in the range of 1 to 3000 inclusive. The test scores are presented as a report in the following form:

PersonCategory <name> s1, s2, s3, ... sN

where PersonCategory is either Adolescent or Volunteer, and each si (1 <= i <= **N**) is an integer value in the range of one (1) to three thousand (3000), inclusive.

The Adolescent Shelter decided to assign weights to indicate the importance of each personality trait. The weights (*w1, w2, … wN*) used for adolescents are different from those weights (*v1, v2, … vN*) used for volunteers. The function:

f1 = (A, V) = (w1 (As1 – Vs1)2 + w2 (As2 – Vs2)2 + ... + wN (AsN – VsN)2

is used to assign a numerical value to the quality of mentoring an adolescent A receives from volunteer V, and the function:

f2 = (A, V) = (v1 (As1 – Vs1)2 + v2 (As2 – Vs2)2 + ... + vN (AsN – VsN)2

is used to assign a numerical value to the quality of experience volunteer V gets from mentoring adolescent A. A smaller value of ***f1*** indicates a higher quality of adolescent mentoring and a smaller value of ***f2*** indicates a higher quality of volunteer experience.

Assuming the names in each category are unique, your task is to write a program to prescribe a pairing of adolescents with volunteers such that no pair would be a better fit with each other than the pairing that your program prescribed for them. That is, no pair of an adolescent ***X*** and a volunteer ***Y*** has both values of ***f1(X, Y)*** and ***f2(X, Y)*** smaller than the values prescribed by your program for both of them.

**The Input**

Input consists of multiple situations. Each situation starts with two integers on a separate line. The first integer **N** (**1 *<=* N *<=* 100**) represents the number of personality traits to be used, and the second integer **P** (**1 *<=* P *<=* 1000**) represents the number of adolescents and also the number of volunteers. The last situation is followed by a line containing two zeros that indicates the end of input data and should not be processed as a valid situation.

The second line contains **N** integers that describe the weights to be used for adolescents. Consecutive integers are separated by a single blank space, and each integer has a value of one (1) and ten (10) inclusive.

The third line contains **N** integers that describe the weights to be used for volunteers. Consecutive integers are separated by a single blank space, and each integer has a value of one (1) and ten (10) inclusive.

The following 2**P** lines describe the reports of 2**P** personality tests. Each such line starts with the PersonCategory followed, after a blank space, by a string with no white spaces that represents the name followed, after a blank space, by **N** integers. The **k***th* integer (**1 <= k <= N**) has a value in the range of one (1) to three thousand (3000) that represents the score for the **k***th* trait. The scores are separated by a blank space.

**The Output**

For each situation, print the situation number (starting with 1, and using the format in the sample) on a separate line. The next **P** lines list the pairing of adolescents and volunteers (using the format in the sample) such that the adolescent names are listed in increasing lexicographic order.

**Sample Input**

2 3

1 1

1 1

Volunteer Smith 20 10

Adolescent Paul 20 30

Adolescent Peter 30 30

Volunteer Kevin 20 40

Volunteer Peter 50 10

Adolescent John 10 20

0 0 1

**Sample Output**

Situation 1:

Adolescent John Volunteer Smith

Adolescent Paul Volunteer Kevin

Adolescent Peter Volunteer Peter