The Game of Master-Mind

If you want to buy a new cellular phone, there are many various types to choose from. To decide which one is the best for you, you have to consider several important things: its size and weight, battery capacity, WAP support, colour, price. One of the most important things is also the list of games the phone provides. Nokia is one of the most successful phone makers because of its famous Snake and Snake II. ACM wants to make and sell its own phone and they need to program several games for it. One of them is Master-Mind, the famous board logical game.

The game is played between two players. One of them chooses a *secret code* consisting of P ordered pins, each of them having one of the predefined set of C colours. The goal of the second player is to guess that secret sequence of colours. Some colours may not appear in the code, some colours may appear more than once.

The player makes guesses, which are formed in the same way as the secret code. After each guess, he/she is provided with an information on how successful the guess was. This feedback is called a *hint*. Each hint consists of B black points and W white points. The black point stands for every pin that was guessed right, i.e. the right colour was put on the right position. The white point means right colour but on the wrong position. For example, if the secret code is "white, yellow, red, blue, white" and the guess was "white, red, white, white, blue", the hint would consist of one black point (for the white on the first position) and three white points (for the other white, red and blue colours). The goal is to guess the sequence with the minimal number of hints.

The new ACM phone should have the possibility to play both roles. It can make the secret code and give hints, but it can also make its own guesses. Your goal is to write a program for the latter case, that means a program that makes Master-Mind guesses.

Input

There is a single positive integer T on the first line of input. It stands for the number of test cases to follow. Each test case describes one game situation and you are to make a guess. On the first line of each test case, there are three integer numbers, P, C and M. P ($1 <= P <= 10$) is the number of pins, C ($1 <= C <= 100$) is the number of colours, and M ($1 <= M <= 100$) is the number of already played guesses.

Then there are 2 x M lines, two lines for every guess. At the first line of each guess, there are P integer numbers representing colours of the guess. Each colour is represented by a number G_i, $1 <= G_i <= C$. The second line contains two integer numbers, B and W, stating for the number of black and white points given by the corresponding hint.

Let's have a secret code S_1, S_2, ... S_P and the guess G_1, G_2, ... G_P. Then we can make a set H containing pairs of numbers (I,J) such that $S_I = G_J$, and that any number can appear at most once on the first position and at most once on the second position. That means for every two different pairs from that set, (I_1,J_1) and (I_2,J_2), we have $I_1 <> I_2$ and $J_1 <> J_2$. Then we denote $B(H)$ the number of pairs in the set, that meet the condition $I = J$, and $W(H)$ the number of pairs with $I <> J$.

We define an ordering of every two possible sets H_1 and H_2. Let's say $H_1 <= H_2$ if and only if one of the following holds:

- $B(H_1) < B(H_2)$, or
- $B(H_1) = B(H_2)$ and $W(H_1) <= W(H_2)$

Then we can find a maximal set H_{max} according to this ordering. The numbers $B(H_{max})$ and $W(H_{max})$ are the black and white points for that hint.

Output

For every test case, print the line containing P numbers representing P colours of the next guess. Your guess must be valid according to all previous guesses and hints. The guess is valid if the sequence could be a secret code, i.e. the sequence was not eliminated by previous guesses and hints.

If there is no valid guess possible, output the sentence You are cheating!. If there are more valid guesses, output the one that is lexicographically smallest. I.e. find such guess G that for every other valid guess V there exists such a number I that:

- $G_J = V_J$ for every $J < I$, and
- $G_I < V_I$.

Example

Sample Input:

```
3
4 3 2
1 2 3 2
1 1
2 1 3 2
1 1
4 6 2
3 3 3 3
3 0
4 4 4 4
2 0
8 9 3
1 2 3 4 5 6 7 8
0 0
2 3 4 5 6 7 8 9
1 0
3 4 5 6 7 8 9 9
2 0
```

Sample Output

```
1 1 1 3
You are cheating!
9 9 9 9 9 9 9 9
```

Warning: large Input/Output data, be careful with certain languages