P7. Placing Lampposts

(Time Limit: 3 seconds)

As a part of the mission ‘Beautification of Dhaka City’, the government has decided to replace all the old lampposts with new expensive ones. Since the new ones are quite expensive and the budget is not up to the requirement, the government has decided to buy the minimum number of lampposts required to light the whole city.

Dhaka city can be modeled as an undirected graph with no cycles, multi-edges or loops. There are several roads and junctions. A lamppost can only be placed on junctions. These lampposts can emit light in all the directions, and that means a lamppost that is placed in a junction will light all the roads leading away from it.

The ‘Dhaka City Corporation’ has given you the road map of Dhaka city. You are hired to find the minimum number of lampposts that will be required to light the whole city. These lampposts can then be placed on the required junctions to provide the service. There could be many combinations of placing these lampposts that will cover all the roads. In that case, you have to place them in such a way that the number of roads receiving light from two lampposts is maximized.

Input

There will be several cases in the input file. The first line of input will contain an integer T ($T \leq 30$) that will determine the number of test cases. Each case will start with two integers N ($N \leq 1000$) and M ($M < N$) that will indicate the number of junctions and roads respectively. The junctions are numbered from 0 to $N - 1$. Each of the next M lines will contain two integers a and b, which implies there is a road from junction a to b, ($0 \leq a, b < N$) and $a \neq b$. There is a blank line separating two consecutive input sets.

Output

For each line of input, there will be one line of output. Each output line will contain 3 integers, with one space separating two consecutive numbers. The first of these integers will indicate the minimum number of lampposts required to light the whole city. The second integer will be the number of roads that are receiving lights from two lampposts and the third integer will be the number of roads that are receiving light from only one lamppost.

Sample Input

```
2
4 3
0 1
1 2
2 3
5 4
```

0 1

0 2

0 3

0 4

Sample Output

2 1 2

1 0 4

P8. Station Balance

(Time Limit: 3 seconds)

The International Space Station contains many centrifuges in its labs. Each centrifuge will have some number (C) of chambers each of which can contain 0, 1, or 2 specimens. You are to write a program which assigns all S specimens to the chambers such that no chamber contains more than 2 specimens and the following expression for *IMBALANCE* is minimized.

$$IMBALANCE = \sum_{i=1}^{C} | CM_i - AM |$$

where:

CM_i is the Chamber Mass of chamber i and is computed by summing the masses of the specimens assigned to chamber i.

AM is the Average Mass of the chambers and is computed by dividing the sum of the masses of all specimens by the number of chambers (C).

Input

Input to this program will be a file with multiple sets of input. The first line of each set will contain two numbers. The first number ($1 \leq C \leq 5$) defines the number of chambers in the centrifuge and the second number ($1 \leq S \leq 2C$) defines the number of specimens in the input set. The second line of input will contain S integers representing the masses of the specimens in the set. Each specimen mass will be between 1 and 1000 and will be delimited by the beginning or end of the line and/or one or more blanks.

Output

For each input set, you are to print a line specifying the set number (starting with 1) in the format 'Set #X' where X is the set number.

The next C lines will contain the chamber number in column 1, a colon in column number 2, and then the masses of the specimens your program has assigned to that chamber starting in column 4. The masses in your output should be separated by exactly one blank.

Your program should then print 'IMBALANCE = X' on a line by itself where X is the computed imbalance of your specimen assignments printed to 5 digits of precision to the right of the decimal. The final line of output for each set should be a blank line. (Follow the sample output format.)

Sample Input

```
2 3
6 3 8
3 5
51 19 27 14 33
5 9
1 2 3 5 7 11 13 17 19
```

Sample Output

```
Set #1
 0: 6 3
 1: 8
IMBALANCE = 1.00000
Set #2
 0: 51
 1: 19 27
 2: 14 33
IMBALANCE = 6.00000
Set #3
 0: 1 17
 1: 2 13
 2: 3 11
 3: 5 7
 4: 19
IMBALANCE = 11.60000
```