GLOBUSSOFT

TECHNOLOGY AHEAD OF TIME

Greetings From Globussoft

- Given below are 5 Programming questions, you have to solve any 3 out of 5 questions.

- These 5 questions you can attempt in any technology like C/C++, java, .Net, PHP

- To solve these 3 questions you've max. 3 hours.

- While Solving these questions you are not allowed to use any Search Engine like Google, Yahoo, Bing …

All the best for your test

Globussoft

QUESTION - 1

Every year there is the same problem at Halloween: Each neighbor is only willing to give a certain total number of sweets on that day, no matter how many children call on him, so it may happen that a child will get nothing if it is too late. To avoid conflicts, the children have decided they will put all sweets together and then divide them evenly among themselves. From last year's experience of Halloween they know how many sweets they get from each neighbour. Since they care more about justice than about the number of sweets they get, they want to select a subset of the neighbours to visit, so that in sharing every child receives the same number of sweets. They will not be satisfied if they have any sweets left which cannot be divided.

Your job is to help the children and present a solution.

Input

The input contains several test cases.
The first line of each test case contains two integers **c** and **n** ($1 \le c \le n \le 100000$), the number of children and the number of neighbours, respectively. The next line contains **n** space separated integers $\mathbf{a_1}$ **, ... ,** $\mathbf{a_n}$ ($1 \le a_i \le 100000$), where $\mathbf{a_i}$ represents the number of sweets the children get if they visit neighbour **i**.

The last test case is followed by two zeros.

Output

For each test case output one line with the indices of the neighbours the children should select (here, index **i** corresponds to neighbour **i** who gives a total number of $\mathbf{a_i}$ sweets). If there is no solution where each child gets at least one sweet, print "no sweets" instead. Note that if there are several solutions where each child gets at least one sweet, you may print any of them.

Example

```
Input:
4 5
1 2 3 7 5
3 6
7 11 2 5 13 17
0 0


Output:
3 5
2 3 4
```

QUESTION – 2

Using a sheet of paper and scissors, you can cut out two faces to form a cylinder in the following way:

1. Cut the paper horizontally (parallel to the shorter side) to get two rectangular parts.
2. From the first part, cut out a circle of maximum radius. The circle will form the bottom of the cylinder.
3. Roll the second part up in such a way that it has a perimeter of equal length with the circle's circumference, and attach one end of the roll to the circle. Note that the roll may have some overlapping parts in order to get the required length of the perimeter.

Given the dimensions of the sheet of paper, can you calculate the biggest possible volume of a cylinder which can be constructed using the procedure described above?

Input Specification

The input consists of several test cases. Each test case consists of two numbers **w** and **h** ($1 \leq w \leq h \leq 100$), which indicate the width and height of the sheet of paper.

The last test case is followed by a line containing two zeros.

Output Specification

For each test case, print one line with the biggest possible volume of the cylinder. Round this number to 3 places after the decimal point.

Sample Input

```
10 10
10 50
10 30
0 0
```

Sample Output

```
54.247
785.398
412.095
```

QUESTION – 3

A cashier in a grocery store seems to have difficulty in distinguishing the multiplication symbol and the addition symbol. To make things easier for him, you want to buy goods in such a way that the product of their prices is the same as the sum of their prices.

Of course, if you buy only one item, this is always true. With two items and three items, it still seems quite a boring task to you, so now you are interested in finding possible prices of four items such that the sum of the four prices is equal to the product of the four prices. You should consider the prices are in € with two digits after the decimal point. Obviously, each product costs at least one cent.

Input Specification

This problem has no input.

Output Specification

Print all solutions which have a sum of the four items of at most **20.00 €**. For each solution, print one line with the prices of the four items in non-decreasing order, with one space character between them. You may print the solutions in any order, but make sure to print each solution only once.

Sample Output

```
0.50 1.00 2.50 16.00
1.25 1.60 1.75 1.84
1.25 1.40 1.86 2.00
...
```

QUESTION – 4

A group of n castle guards are voting to determine whether African swallows can carry coconuts. While each guard has his own personal opinion on the matter, a guard will often vote contrary to his beliefs in order to avoid disagreeing with the votes of his friends.

You are given a list of guards who either do or do not believe in the coconut-carrying capacity of African swallows, and a list of all pairs of guards who are friends. Your task is to determine how each guard must vote in order to minimize the sum of the total number of disagreements between friends and the total number of guards who must vote against their own beliefs.

Input

The input to this problem will contain multiple test cases. Each test case begins with a single line containing an integer n (where 2 <= n <= 300), the number of guards, and an integer m (where 1 <= m <= n(n-1)/2), the number of pairs of guards who are friends. The second line of the test case contains n integers, where the ith integer is 1 if the ith guard believes in the ability of African swallows to carry coconuts, and 0 otherwise. Finally, the next m lines of the test case each contain two distinct integers i and j (where 1 <= i, j <= n), indicating that guards i and j are friends. Guards within each pair of friends may be listed in any order, but no pair of guards will be repeated. The input is terminated by an invalid test case with n = m = 0, which should not be processed.

Output

For each input test case, print a single line containing the minimum possible sum of the total number of disagreements between all friends plus the total number of guards who must vote against their own beliefs.

Example

```
Input:
3 3
1 0 0
1 2
1 3
3 2
6 6
1 1 1 0 0 0
1 2
2 3
4 2
3 5
4 5
5 6
0 0

Output:
1
2
```

QUESTION – 5

Ann Sister owns a genealogical database service, which maintains family tree history for her clients. When clients login to the system, they are presented with a variety of services: searching, printing, querying, etc. One recent question that came up which the system was not quite prepared for was the following: “Which member of my family had the most grandchildren?” The client who posed this question eventually had to answer it by manually searching the family tree database herself. Ann decided to have software written in case this question (or ones similar to it asking for great-grandchildren, or great-great-grandchildren, etc.) is asked in the future.

Input

Input will consist of multiple test cases. The first line of the input will contain a single integer indicating the number of test cases. Each test case starts with a single line containing two positive integers n and d, where n indicates the number of lines to follow containing information about the family tree, and d indicates the specific question being asked about the tree: if d = 1, then we are interested in persons with the most children (1 generation away); if d = 2, then we are interested in persons with the most grandchildren (2 generations away), and so on. The next n lines are of the form

name m dname$_1$ *dname*$_2$ ... *dname*$_m$

where *name* is one of the family members' names, *m* is the number of his/her children, and *dname*$_1$ through *dname*$_m$ are the names of the children. These lines will be given in no particular order. You may assume that all n lines describe one single, connected tree. There will be no more than 1000 people in any one tree, and all names will be at most 10 characters long.

Output

For each test case, output the three names with the largest number of specified descendants in order of number of descendants. If there are ties, output the names within the tie in alphabetical order. Print fewer than three names if there are fewer than three people who match the problem criteria (you should not print anyone's name who has 0 of the specified descendants), and print more than three if there is a tie near the bottom of the list. Print each name one per line, followed by a single space and then the number of specified descendants. The output for each test case should start with the line

Tree *i*:

where *i* is the test case number (starting at 1). Separate the output for each problem with a blank line.

Example

Input:

```
3
8 2
Barney 2 Fred Ginger
Ingrid 1 Nolan
Cindy 1 Hal
Jeff 2 Oliva Peter
Don 2 Ingrid Jeff
Fred 1 Kathy
Andrea 4 Barney Cindy Don Eloise
Hal 2 Lionel Mary
6 1
Phillip 5 Jim Phil Jane Joe Paul
Jim 1 Jimmy
Phil 1 Philly
Jane 1 Janey
Joe 1 Joey
Paul 1 Pauly
6 2
Phillip 5 Jim Phil Jane Joe Paul
Jim 1 Jimmy
Phil 1 Philly
Jane 1 Janey
```

```
Joe 1 Joey
Paul 1 Pauly
```

Output:

```
Tree 1:
Andrea 5
Don 3
Cindy 2
Tree 2:
Phillip 5
Jane 1
Jim 1
Joe 1
Paul 1
Phil 1
Tree 3:
Phillip 5
```