The Sweet Job

As all children of his age, little Johnny was in love with sweets. One day he decided to earn his treat and found a job in the Research & Development department of his local grocery store. Every morning, the delivery boys (milkmen and papermen) employed by the store go on their rounds: collected goods from a fixed depot, visited some number of houses along their route, and finally returned to the depot when their work was complete.

Johnny's first assignment is developing a cost-cutting scheme for the home delivery services provided by the store. It is his task to select a location for the depot, and individual routes for all delivery men, so as to minimise the total distance covered, and guarantee that each house is included in some delivery man's round. Since a meeting between two employees would undoubtedly result in a major delay (due to their lazy nature), it is required that the routes of any two delivery men are to have only one point in common (the depot). The store manager is not particularly attached to his workers, so if some of the delivery men find themselves jobless (without a round), no harm is done.

Input

The first line of input contains a single integer *t*, the number of test cases (*t* = 1000). *t* test cases follow.

Each test case starts with a line containing two integers *n k*, denoting the number of houses and the number of children, respectively (1 <= *k* <= 16, 1 <= *n* <= 256). Each of the next *n* lines contains two integers x_i y_i each (-1000 <= x_i,y_i <= 1000), denoting the coordinates of the houses.

Output

For the *i*-th test case output a line with the text `case i Y` or `case i N`, specifying whether you wish to solve the given case. Then in the former case print exactly *k* lines. Each line should start with integer *p* <> 1 and be followed by a space separated list of exactly *p* integers, representing the houses (numbered in input order) visited by the *p*-th delivery man. Delivery men are assumed to move along straight line segments between houses.

Score

The score awarded to your program is the total of scores for the test cases you chose to solve.

For each solved test case you will receive *diam* / *d* points, where *diam* denotes the distance between the two furthest houses, and *d* is the total distance of routes of all delivery men.

Example

Input:

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

```
3 0
```

Output:

```
case 1 Y
2 1 2
2 3 4
0
```

Score:

```
0.750001
```

Bonus info: If score = *xxx.xxxaaa*, *aaa* means the number of test cases with Y answer.