

The Secret Fellowship of Byteland

The relationship between The University of Byteland and King Johnny was never a friendly one. The king was the easy-going, open-minded sort of person who is prepared to turn a blind eye to the embezzlement of public funds, but inwardly revolts at the thought of money going to waste, and supporting a university was to the king a perfect example of a waste of money. On the other hand, the chancellor of the university showed no tolerance whatsoever, and frequently stated in public that Byteland was being governed by a monarch who took terrible decisions when he was drunk and even worse ones when he was sober. After some time of bad-tempered coexistence, the king had had enough and decided to close down the university. However, the king's councillors advised against this move, suggesting it might cause social unrest. The king yielded to their advice, and instead established a law which banned all organisations, clubs and associations active at the university.

This action had a rather curious effect on the usually lazy students of the university. They had never before even thought of organising any sort of fellowship, but now they immediately decided they needed to set one up. And this is how the *Secret Fellowship* came to life.

The main problem faced by the management of the Fellowship was how to organise members' meetings in such a way as to minimise the risk to the participants. It was decided that the n members of the fellowship should be split into k secret divisions, each consisting of at least 2 members. All members belonging to the same division would then meet regularly, and they would take it in turns to host the meetings of the division in their houses.

But one more important factor has to be taken into account -- the laziness of students. It is therefore your task to form the divisions in such a way that the furthest distance a student may ever be asked to walk is as short as possible.

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 students and the number of divisions to be formed, respectively ($2 \leq k \leq n \leq 200$). Each of the next n lines contains two integers x_i y_i each ($-1000 \leq x_i, y_i \leq 1000$), denoting the coordinates of the houses of successive students.

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 n_j ($n_j \geq 2$) and be followed by a space separated list of exactly n_j increasing integers s_{j1} , denoting the students belonging to the j -th division, numbered in input order ($1 \leq s_{j1} \leq n$). All divisions must be disjoint and the sum of all numbers n_j must equal n .

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 $diam / (d^k)$ points, where $diam$ denotes the distance between the two furthest houses of members of the fellowship, and d is the distance between the two furthest houses of members belonging to the same division.

Example

Input:

```
3
6 2
0 0
1 0
0 1
1 1
2 0
2 1
6 2
0 0
1 0
0 1
1 1
2 0
2 1
6 2
0 0
1 0
0 1
1 1
2 0
2 1
```

Output:

```
case 1 Y
3 1 2 4
3 3 5 6
case 2 Y
3 1 2 5
3 3 4 6
case 3 Y
2 1 3
4 2 4 5 6
```

Score:

```
1.849003
```

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