Herdkeeper

After the tragic end of the watermelon plantation, Johnny has switched to farming. More precisely, he is now a Certified Livestock Supervisor (i.e. shepherd) tending herds of antelope. It is his task to divide the animals into herds, and to build a fence around each herd, trying to keep the total length of all fences as small as possible. Each herd must consist of at least 2 antelope, and the antelope may stand arbitrarily close to the fence itself.

Input

t [the number of test cases <= 1000]
n [2 <= the number of antelope <= 100]
x_1 y_1[coordinates of the antelope, between -1000 and 1000]
x_2 y_2
.....
x_n y_n
[next test cases]

Output

case *i* Y [N - if you want to skip the testcase]
c [the number of herds]
*a s*1 *s*2 ... s_a [2 <= *a* - the number of antelope in the first herd, and the numbers of the antelope in this herd in the order from the input sequence]
[next test cases]

Scoring

The score awarded to your program is the sum of scores for individual test cases. For the i-th test case you will receive 1 / (1 + *sum* / *conv*) points, where *sum* is the sum of circumferences of all convex hulls of herds in your solution, and the *conv* is the circumference of the convex hull of the set of all antelope. If you don't want to solve the *i*-th test case, you may output the line 'case *i* N' and nothing else.

Example

Input:

```
6
2
0 0
5 0
3
4 0
-4 -5
2 3
5
```

```
20 10
10 10
40 50
-20 -40
-30 -20
4
2 4
2 -4
2 0
-5 -3
3
2 4
-4 -4
2 3
4
-1 -3
-1 5
3 -5
-1 5
```

Output:

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

Score:

```
3.079001
```

Bonus info: If score = *xxx.xxxaaa*, *aaa* means the number of test cases with score > 0.5