

Primitivus recurrence

A **genetic code** of the abstract primitivus (*Primitivus recurrence*) is a series of natural numbers $K=(a_1, \dots, a_n)$. A **feature** of primitivus we call each ordered pair of numbers (l, r) , which appears successively in the genetic code, i.e. there exists such i that $l=a_i, r=a_{i+1}$. There are no (p, p) features in a primitivus' genetic code.

Task

Write a program which:

- reads the list of the features from the standard input,
- computes the length of the shortest genetic code having given features,
- writes the results to the standard output.

Input

The number of test cases t is in the first line of input, then t test cases follow separated by an empty line. In the first line of each test case one positive integer number n is written. It is the number of different features of the primitivus. In each of the following n lines there is a pair of natural numbers l and r separated by a single space, $1 \leq l \leq 1000, 1 \leq r \leq 1000$. A pair (l, r) is one of the primitivus' features. The features do not repeat in the input file

Output

Your program should write for each test case exactly one integer number equal to the length of the shortest genetic code of the primitivus, comprising the features from the input.

Example

Sample input:

```
1
12
2 3
3 9
9 6
8 5
5 7
7 6
4 5
5 1
1 4
4 2
2 8
8 6
```

Sample output:

```
15
```

All the features from the example are written in the following genetic code:

```
(8, 5, 1, 4, 2, 3, 9, 6, 4, 5, 7, 6, 2, 8, 6)
```

Warning: enormous Input/Output data, be careful with certain languages