Problem: Badges

A badge is created by superimposing two squares. The bottom square should be of blue colour and top square should be of yellow colour. The squares must be rotated in such a way that the projection of the badge displays 8 distinguishable vertices. Please see following image for an example. In the picture, the left-most configuration is permitted while the other two are not.
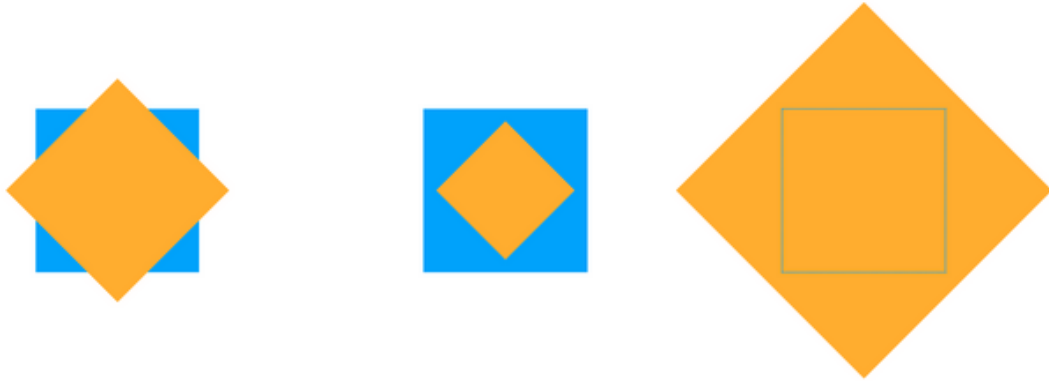
You are given n squares, each of a given color and side length. Determine the maximum number of badges that you can make. Note that each square can be used for at most one badge.

Input Format:

The first line of input file contains integer n. Each of the next **n** lines contain two space separated integers, c_i and d_i. c_i is the colour number (1 for blue and 2 for yellow) and d_i is the length of side of square.

Output Format:

Print the maximum number of badges that can be made.

Constraints:

$1 <= n <= 100000$

$1 <= d_i <= 1000$

$1 <= c_i <= 2$

Sample Input

```
10
1 3
1 17
2 13
2 13
2 6
1 10
2 8
1 7
1 8
2 8
```

Sample Output

```
4
```

Resources:

Try: https://www.spoj.com/problems/MATCHING/

Dinic's algorithm: https://www.youtube.com/watch?v=M6cm8UeeziI

Dinic's Implementation: (Java) https://www.youtube.com/watch?v=_SdF4KK_dyM&t=0s

(C++) https://cp-algorithms.com/graph/dinic.html