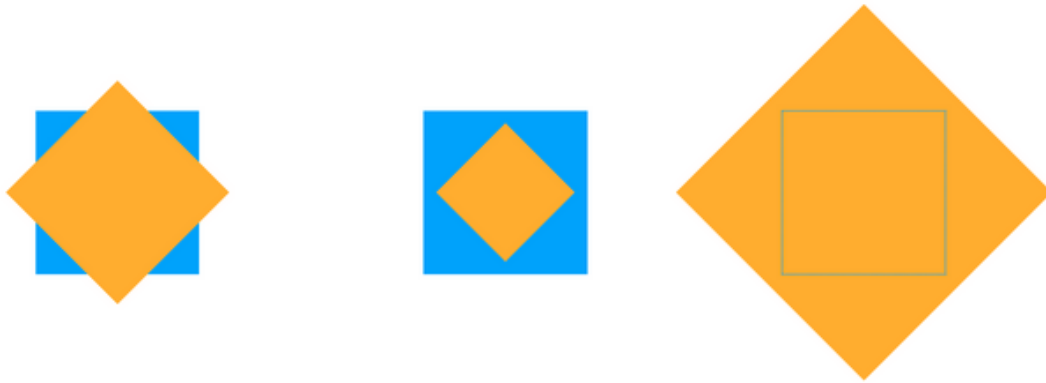


Getting started with Competitive Programming

Week 8 – Network Flow

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:

$$1 \leq n \leq 100000$$

$$1 \leq d_i \leq 1000$$

$$1 \leq c_i \leq 2$$

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>

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
```

Test case 1

```
10
2 8
1 15
2 11
1 8
1 2
2 1
1 17
1 6
1 18
2 5
```

Output: 3

Test case 2

```
10
1 13
1 12
1 6
2 19
2 19
1 18
1 7
```

2 10
1 12
2 10

Output: 3