```
#!/bin/python3

import math
import os
import random
import re
import sys

#
# Complete the 'spot_the_y' function below.
#
# The function is expected to return an INTEGER.
# The function accepts following parameters:
#  1. INTEGER n
#  2. STRING line
#

def spot_the_y(n, line):
    board = ["-" for x in range(n**2)]
    count = 0
    x = line.split()
    for l in x:
        if count>7:
            break
        if count%2 == 1:
            if board[int(l)-1] == "-":
                board[int(l)-1] = "$"
            else:
                board[int(l)-1] = "-"
```

```
        else:
            if board[int(l)-1] == "-":
                board[int(l)-1] = "@"
            else:
                board[int(l)-1] = "-"
        count+=1
    while check_for_y(board)==0 and count<len(x):
        if count%2==0:
            if board[int(x[count])-1] == "-":
                board[int(x[count])-1] = "@"
            else:
                board[int(x[count])-1] = "-"
        else:
            if board[int(x[count])-1] == "-":
                board[int(x[count])-1] = "$"
            else:
                board[int(x[count])-1] = "-"
        count+=1
    return check_for_y(board)

def check_for_y(board):
    p1list = []
    p2list = []
    n_temp = int(len(board)**0.5)
    for i in range(len(board)):
        if board[i] == "@":
            p1list.append(i)
        elif board[i] == "$":
            p2list.append(i)
```

```
for d in p1list:
    if d%n_temp != 0 and d%n_temp != n_temp-1 and d>n_temp-1 and d<=n_temp*(n_temp-1)-1:
        if d-1 in p1list:
            if d-n_temp+1 in p1list and d+n_temp+1 in p1list:
                return int(d+d-1+d-n_temp+1+d+n_temp+1+4)
        if d+1 in p1list:
            if d-n_temp-1 in p1list and d+n_temp-1 in p1list:
                return int(d+d+1+d-n_temp-1+d+n_temp-1+4)
        if d-n_temp in p1list:
            if d+n_temp-1 in p1list and d+n_temp+1 in p1list:
                return int(d+d-n_temp+d+n_temp-1+d+n_temp+1+4)
        if d+n_temp in p1list:
            if d-n_temp-1 in p1list and d-n_temp+1 in p1list:
                return int(d+n_temp+d+d-n_temp-1+d-n_temp+1+4)
for b in p2list:
    if b%n_temp != 0 and b%n_temp != n_temp-1 and b>n_temp-1 and b<=n_temp*(n_temp-1)-1:
        if b-1 in p2list:
            if b-n_temp+1 in p2list and b+n_temp+1 in p2list:
                return int(b+b-1+b-n_temp+1+b+n_temp+1+4)
        if b+1 in p2list:
            if b-n_temp-1 in p2list and b+n_temp-1 in p2list:
                return int(b+b+1+b-n_temp-1+b+n_temp-1+4)
        if b-n_temp in p2list:
            if b+n_temp-1 in p2list and b+n_temp+1 in p2list:
                return int(b+b-n_temp+b+n_temp-1+b+n_temp+1+4)
        if b+n_temp in p2list:
            if b-n_temp-1 in p2list and b-n_temp+1 in p2list:
                return int(b+n_temp+b+b-n_temp-1+b-n_temp+1+4)
return 0
```

```
if __name__ == '__main__':
    fptr = open(os.environ['OUTPUT_PATH'], 'w')

    n = int(input().strip())

    line = input()

    result = spot_the_y(n, line)

    fptr.write(str(result) + '\n')

    fptr.close()
```