JustinBieber – BABY

Source: ACM-ICPC Kanpur Site 2007

A baby tries to solve the well-known eight-queen puzzle: the problem of placing eight chess queens on an 8×8 chessboard so that no two queens share the same row, column, or diagonal. The baby understands the concept of row and column quite well but diagonal is not very clear to her. As a result she succeeds placing eight queens on the board so that no two queens share the same row or column but there remains the possibility that some queens share the same diagonal.

Given baby's queens (a solution by the baby) and a valid eight-queen solution, it is possible to move baby's queens to positions of queens in the valid solution. Assume that in a single move, a queen can be moved one unit row-wise or column-wise into an unoccupied position.

Write a program to find the minimum number of moves required to move baby's queens to positions of queens in the valid solution. The program should be usable for a more general n -queen puzzle where n queens are placed on an n×n chessboard, $4 \le n \le 16$. Assume that rows and columns of the chessboard are numbered 1, 2,..., n .

Input: JustinBieber.inp

- The input consists of multiple test cases.
- Each case begins with a line containing the integer n .
- Each of the next two lines contains a sequence of n integers. Integers in the first line represent column numbers of baby's queens appearing in rows 1, 2,..., n respectively. In the same way, the second line contains column numbers of queens in the given valid solution. A space separates two consecutive integers in the sequence.
- A line containing a zero.

Output: JustinBieber.out

- For each test case, print the minimum number of moves required.

Example

Sample Input

```
4
1 2 3 4
3 1 4 2
4
3 2 4 1
3 1 4 2
5
5 3 1 4 2
5 3 1 4 2
5
1 5 2 4 3
3 1 4 2 5
0
```

Sample Output

```
6
2
0
8
```