Chess is a very popular game in the West. In Chess, queen is a powerful unit due to its movement. The queen has three kinds of movement: Vertical, Horizontal, and Diagonal. Along the three directions, there is no limit to the moving distance. The eight queen puzzle is to find an arrangement of eight queens so that queens will not attack each other. There is only one queen in each row There is only one queen in each column There is at most one queen in each diagonal.

There are two common representations for the eight queen puzzle: matrix and permutation. In the matrix representation, 0 as empty and 1 as queen for each cell. As for the representation of permutation, the index is the x coordinate and the value of the element is the y coordinate for the eight queens. For example, a four-queen matrix can be:

1 0 0 0
0 0 1 0
0 0 0 1
0 1 0 0

and its corresponding permutation is:

0 2 3 1

If any two queens will not reside in the same row and column, then an attack only occurs when the difference of x coordinate equals to the difference of y coordinate. Thus, in the above example we have one attack.

Your goal is to write a program to translate the matrix form to a permutation form and calculate the number of attacks. You have to define a class with an constructor for binary matrix input, two private data members to store the permutation and number of attacks, and two methods to calculate the number of attacks, and to print the permutation and the result of number of attacks.

Please overload the stream insertion operator to input the matrix to the class object:

cin >> eightQueensObject;

And make the calculation of number of attacks be possible to make cascaded call:

eightQueensObject.calculate_attacks().print();

Requirement: Provide a class satisfying the above requirements. Prepare your class with appropriate constructor and encapsulate the required methods. Separate your program in three files: the class header file (.h), the class source code file (.cpp), and the file containing main function (.cpp).

Prohibited: Use C-style input/output.

Input

Each case contains an 8-by-8 matrix. The input ends with -1.

Output

For each case, use the matrix to construct the class object, and output the corresponding permutation and number of attack. The permutation numbers are separated by space and end with a colon, followed by a space, the number of attacks, and an end line stream manipulator.

Sample Input

0 1 0 0 0 0 0 0
0 0 1 0 0 0 0 0
0 0 0 0 0 0 1 0
0 0 0 1 0 0 0 0
0 0 0 0 0 1 0 0
1 0 0 0 0 0 0 0
0 0 0 0 1 0 0 0
0 0 0 0 0 0 0 1
-1

Sample Output

1 2 6 3 5 0 4 7: 4 Attacks