/\*

You are given an n x n 2D matrix representing an image.

Rotate the image by 90 degrees (clockwise).

Follow up:

Could you do this in-place?

方法一：纯找数学规律，按照相应应该在的位置安放

方法二：

本地使得二维矩阵，旋转90角度。

通过实际数据分析，通过两个步骤的元素交换可实现目标：

1\按照主对角线，将对称元素交换

2\按照列，将对称列元素全部交换

即可达到，使得二维矩阵，本地旋转90个角度。

\*/

class Solution {

public:

void jiaohuan(vector<vector<int>>& matrix,int i,int h)

{

int n=matrix.size();//4

int temp;

temp=matrix[i][h];//0 1

matrix[i][h]=matrix[n-1-h][i];//20 ->01

matrix[n-1-h][i]=matrix[n-1-i][n-1-h];// 32 ->20

matrix[n-1-i][n-1-h]=matrix[h][n-1-i];

matrix[h][n-1-i]=temp;

}

void rotate(vector<vector<int>>& matrix)

{

//way-1

/\*

int n=matrix.size();

int h,cishu;

for(int i=0;i<n/2;i++)

{

h=i;

cishu=(n-1)-2\*i;

while(cishu>0)

{

jiaohuan(matrix,i,h);

h++;

cishu--;

}

}

\*/

//way-2

int n=matrix.size();

for(int i=0;i<n;i++)

for(int j=0;j<i;j++)

swap(matrix[i][j],matrix[j][i]);

for(int i=0;i<n;i++)

for(int j=0;j<n/2;j++)

swap(matrix[i][j],matrix[i][n-1-j]);

}

};