/\*

Given a matrix of m x n elements (m rows, n columns), return all elements of the matrix in spiral order.

For example,

Given the following matrix:

[

[ 1, 2, 3 ],

[ 4, 5, 6 ],

[ 7, 8, 9 ]

]

You should return [1,2,3,6,9,8,7,4,5].

旋转读取二维矩阵

方法一：读取之后删除，然后就小了一圈，迭代读取

方法二：不同迭代，外围套for循环也可以

\*/

class Solution {

public:

void guocheng(vector<vector<int>>& matrix,vector<int> &result)

{

int lie=matrix[0].size();//列数

if(lie==1)//如果只有一列

{

for(int i=0;i<matrix.size();i++)

result.push\_back(matrix[i][0]);

return;

}

for(int i=0;i<lie;i++)//第一行输进去

result.push\_back(matrix[0][i]);

matrix.erase(matrix.begin());//删除第一行

int hang=matrix.size();//右边一竖进去

if(hang==0)//如果只有一行

return;

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

{

result.push\_back(matrix[i][lie-1]);

matrix[i].erase(matrix[i].begin()+lie-1);//挨个删除右边一列

}

lie=matrix[0].size();//重新计算列数

if(lie==1)

{

for(int i=matrix.size()-1;i>=0;i--)

result.push\_back(matrix[i][0]);

return;

}

for(int i=lie-1;i>=0;i--)//最后一行输进去

result.push\_back(matrix[hang-1][i]);

matrix.erase(matrix.begin()+hang-1);//删除最后一行

hang=matrix.size();//重新计算行数

for(int i=hang-1;i>=0;i--)

{

result.push\_back(matrix[i][0]);

matrix[i].erase(matrix[i].begin());//挨个删除左边一列

}

if(matrix.size()>0)

guocheng(matrix,result);

else

return;

return;

}

vector<int> spiralOrder(vector<vector<int>>& matrix)

{

vector<int> result;

if(matrix.size()==0)

return result;

guocheng(matrix,result);

return result;

}

};