一、题目说明

题目221. Maximal Square，给一个0和1组成的矩阵，计算包括1的最大正方形。

二、我的解答

这个题目“似曾相识”，用dp可以解决。`dp[i][j]`表示从该节点到右下角，连续正方形的边长：

```
class Solution{
    public:
        int maximalSquare(vector<vector<char>>& matrix){
            if(matrix.empty()) return 0;
            
            int row = matrix.size();
            int col = matrix[0].size();
            vector<vector<int>> dp(row,vector<int>(col,0));
            int maximal = 0;
            for(int i=row-1;i>=0;i--){
                for(int j=col-1;j>=0;j--){
                    if(matrix[i][j]=='0'){
                        dp[i][j] = 0;
                    }else{
                        if(i==(row-1) || j==(col-1)){
                            dp[i][j] = 1;
                            if(dp[i][j]>maximal) maximal = dp[i][j];
                        }else{
                            dp[i][j] = 1 + min(min(dp[i+1][j],dp[i][j+1]),dp[i+1][j+1]);
                            if(dp[i][j]>maximal) maximal = dp[i][j];
                        }
                    }
                }
            }
            return maximal*maximal;
        }
};
```

性能如下：

```
Runtime: 20 ms, faster than 81.76% of C++ online submissions for Maximal Square.
Memory Usage: 11.1 MB, less than 62.96% of C++ online submissions for Maximal Square.
```

三、优化措施

无