

原 给定一个矩阵m，从左上角开始每次只能向右走或者向下走，最后达到右下角的位置

2018年04月27日 21:22:42 阅读数：149

```
public class MaritixMin {  
    /*  
     * 给定一个矩阵m，从左下角开始每次只能向右走或者向下走，最后达到右下角的位置。  
     * 路径中所有数字累加起来就是路径的和，返回所有路径的最小路径和。  
     * dp[i][j]表示的是从原点到i，j位置的最短路径和，我们首先计算第一列和第一行，直接累加即可  
     * 那么对于其他位置，要么从它左边位置达到，要么是从上边的位置达到，我们取左边和上边的较小值，  
     * 然后加上当前的路径值。  
     */  
    public static int minDis(int[][] m) {  
        int [][] dp = new int[4+1][4+1];  
        dp[0][0] = m[0][0];  
        for (int i = 1; i < m.length; i++) {  
            for (int j = 1; j < m[0].length; j++) {  
                if(i==1) {  
                    dp[i][j] = dp[i][j-1]+m[i][j];  
                }else if(j==1) {  
                    dp[i][j] = dp[i-1][j] +m[i][j];  
                }else {  
                    int temp1 = dp[i][j-1]+m[i][j];  
                    int temp2 = dp[i-1][j]+m[i][j];  
                    int min = temp1<temp2?temp1:temp2;  
                    dp[i][j] = min;  
                }  
            }  
        }  
        return dp[m[0].length-1][m.length-1];  
    }  
    public static void main(String[] args) {  
        int[][] martix = {{0,0,0,0,0},  
                           {0,1,3,5,9},  
                           {0,8,1,3,5},  
                           {0,5,0,6,1},  
                           {0,8,8,4,0}};  
        System.out.println("最右下的最短路径为: "+minDis(martix));  
    }  
}
```