**SPIRAL MATRIX:**

**Example 1:**



**Input:** matrix = [[1,2,3],[4,5,6],[7,8,9]]

**Output:** [1,2,3,6,9,8,7,4,5]

**ONLY APPROACH:**

**-> iterate from left to right (top++)**

**-> iterate from top right to top bottom(right--)**

**->iterate from bottom right to bottom left(bottom--)**

**->iterate from bottom left to top left (left++)**

class Solution {

public List<Integer> spiralOrder(int[][] m) {

int top = 0;

int right = m[0].length-1;

int left = 0;

int bottom = m.length-1;

List<Integer> res = new ArrayList();

while(left<=right && top<=bottom)

{

**for(int i = left; i<=right;i++)//left to right**

**res.add(m[top][i]);**

**top++;**

**for(int i= top; i<=bottom;i++)//right corner to bottom right**

**res.add(m[i][right]);**

**right--;**

**if(top<=bottom)**

**{**

**for(int i = right; i >= left; i--)**

**res.add(m[bottom][i]);**

**bottom--;**

**}**

**if(left<=right){**

**for(int i = bottom ; i>=top;i--)**

**res.add(m[i][left]);**

**left++;**

}

}

return res;}}