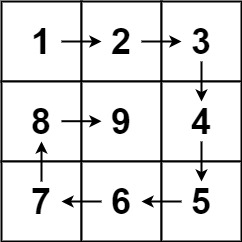
# Generate Spiral Matrix

Given a positive integer n, generate an n x n matrix filled with elements from 1 to n2 in spiral order.

**Example 1:**



**Input:** n = 3

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

**Example 2:**

**Input:** n = 1

**Output:** [[1]]

**Constraints:**

* 1 <= n <= 20

#include <iostream>

#include <iomanip>

#include <vector>

using namespace *std*;

void display2DMatrix(*vector*<*vector*<int>>&& matrix)

{

for (auto i : matrix)

{

for (auto j : i)

{

*cout* << *setw*(2) << j << " ";

}

*cout* << *endl*;

}

*cout* << *endl*;

}

void displayVector(const *vector*<int>& i)

{

for (auto j : i)

{

*cout* << j << " ";

}

*cout* << *endl*;

}

*vector*<*vector*<int>> generateMatrix(int n) {

*vector*<*vector*<int>> matrix(n,*vector*<int>(n));

int minrow = 0, mincol = 0;

int maxrow = n - 1, maxcol = n - 1;

int numberOfElements = n \* n;

int element = 1;

while (element <= numberOfElements)

{

for (auto i = minrow, j = mincol; j <= maxcol && element <= numberOfElements; j++)

{

matrix[i][j] = element;

element++;

}

minrow++;

for (auto i = minrow, j = maxcol; i <= maxrow && element <= numberOfElements; i++)

{

matrix[i][j] = element;

element++;

}

maxcol--;

for (auto i = maxrow, j = maxcol; j >= mincol && element <= numberOfElements; j--)

{

matrix[i][j] = element;

element++;

}

maxrow--;

for (auto i = maxrow, j = mincol; i >= minrow && element <= numberOfElements; i--)

{

matrix[i][j] = element;

element++;

}

mincol++;

}

return matrix;

}

int main(void)

{

//vector<vector<int>> matrix{ {1, 2, 3},{4, 5, 6},{7, 8, 9} };

display2DMatrix(*std*::*move*(generateMatrix(3)));

return 0;

}