1 . FizzBuzz

class Solution {

public:

vector<string> fizzBuzz(int n) {

vector<string> ans;

for(int i=1; i<=n; i++)

{

if(i%3==0 && i%5==0)

{

ans.push\_back("FizzBuzz");

}

else if(i%3==0)

{

ans.push\_back("Fizz");

}

else if(i%5==0)

{

ans.push\_back("Buzz");

}

else

{

ans.push\_back(to\_string(i));

}

}

return ans;

}

};

2 . Pascal triangle

class Solution {

public:

vector<vector<int>> generate(int numRows) {

vector<vector<int>> res;

vector<int> temp;

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

{

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

{

if(j==0 || j==i)

{

temp.push\_back(1);

}

else

{

temp.push\_back(res[i-1][j-1] + res[i-1][j]);

}

}

res.push\_back(temp);

temp.clear();

}

return res;

}

};

3. Spiral Order

class Solution {

public:

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

int n = matrix.size(); // rows

int m = matrix[0].size(); // columns

int sr = 0;

int sc = 0;

int er = n-1;

int ec = m-1;

vector<int> res;

while(sr<=er && sc<=ec)

{

for(int i=sc; i<=ec; i++)

{

res.push\_back(matrix[sr][i]);

}

sr++;

for(int i=sr; i<=er; i++)

{

res.push\_back(matrix[i][ec]);

}

ec--;

if(sr<=er)

{

for(int i=ec; i>=sc; i--)

{

res.push\_back(matrix[er][i]);

}

}

er--;

if(sc<=ec)

{

for(int i=er; i>=sr; i--)

{

res.push\_back(matrix[i][sc]);

}

}

sc++;

}

return res;

}

};

4. Reverse Integer

class Solution {

public:

int reverse(int x) {

long long int ans = 0;

int rem;

bool sign=false;

if(x < 0)

{

sign = true;

x = abs(x);

}

while(x>0)

{

rem = x%10;

ans = ans\*10 + rem;

x /= 10;

}

if(ans > INT\_MAX || ans < INT\_MIN)

{

return 0;

}

if(sign)

{

ans = ans\*(-1);

}

return ans;

}

};