Problems ->

const map1 = new Map();

map1.set('a', 1);

map1.set('b', 2);

map1.set('c', 3);

console.log(map1.get('a'));

// Expected output: 1

map1.set('a', 97);

console.log(map1.get('a'));

// Expected output: 97

console.log(map1.size);

// Expected output: 3

map1.delete('b');

console.log(map1.size);

// Expected output: 2

Map -> Set , get , size , delete , clear, has , keys , values, entries

Set -> add, has , delete , size, clear, keys , values ,entries

**Check Palindrome – II**

module.exports = {

//param A : string

//return an integer

solve : function(A){

var map = new Map();

for(var i=0; i< A.length ; i++) {

if(A[i] in map) {

map[A[i]]++;

}

else {

map[A[i]] = 1;

}

}

var odd =0;

for(val in map) {

if(map[val] %2 === 0) {

continue

}

else {

odd++;

}

}

if(odd >1) {

return 0

}

else {

return 1

}

}

};

Factorial ->

**Find Fibonacci – II**

**Example Explanation**

Explanation 1:

f(2) = f(1) + f(0) = 1

Explanation 2:

f(9) = f(8) + f(7) = 21 + 13 = 34

public class Solution {

public int findAthFibonacci(int A) {

if(A <= 1) {

return A;

}

return findAthFibonacci(A-2) + findAthFibonacci(A-1);

}

}

**Q3. Merge Two Sorted Arrays**

module.exports = {

//param A : array of integers

//param B : array of integers

//return a array of integers

solve : function(A, B){

var results =[];

var i=0;

var j=0;

while(i< A.length && j< B.length) {

if(A[i] > B[j]) {

results.push(B[j]);

j++;

}

else {

results.push(A[i]);

i++;

}

}

while(i< A.length) {

results.push(A[i]);

i++;

}

while(j< B.length) {

results.push(B[j]);

j++;

}

return results;

}

};

**Q2. Sum of all Submatrices**

module.exports = {

//param A : array of array of integers

//return an integer

solve : function(A){

var rows= A.length;

var cols = A[0].length;

var count = 0;

for(var i=0; i<rows; i++) {

for(var j=0; j< cols; j++) {

count += (i+1) \* (j+1 )\* (rows - i ) \* (cols - j) \* A[i][j];

}

}

return count

}

};