1. Explore more applications of Catalan numbers.

(Hint: Grid, Dynamic Programming, Parenthesis)

**Programming Language: Java**

**Code by Chinmai:**

class Solution {

public int numTrees(int n) {

//base case condition

if(n==0 || n==1)

return 1;

int sum=0,i=0,j=n-1;

while(i<=n-1 && j>= 0){

sum += numTrees(i)\*numTrees(j);

i++;

j--;

}

return sum;

}

}

**Code with Memoization Approach:**

public Map<Integer,Integer> hash = new Hashtable<>();

public int numTrees(int n) {

// memoization is working

if(hash.get(n) != null)

return hash.get(n);

// Base case condition

if(n==0 || n==1)

return 1;

int sum=0,i=0,j=n-1;

while(i<=n-1 && j>= 0){

sum += numTrees(i)\*numTrees(j);

i++;

j--;

}

//storage corresponding to n is happening

hash.put(n,sum);

return sum;

}

Count Sort by Ankan:

function countingSort(arr) {

const countingArray = new Array(arr.length + 1);

for(let i=0; i<countingArray.length; i++){

countingArray[i] = 0;

}

for(let num of arr){

countingArray[num] += 1;

}

for(let i=1;i<countingArray.length;i++){

countingArray[i] += countingArray[i-1];

}

const sortedArray = new Array(arr.length);

for(let unsortedArrayPos=arr.length-1; unsortedArrayPos>=0; unsortedArrayPos--){

const sortedArrayPos = countingArray[arr[unsortedArrayPos]];

countingArray[arr[unsortedArrayPos]] -= 1;

sortedArray[sortedArrayPos - 1] = arr[unsortedArrayPos];

}

return sortedArray;

}

console.log(countingSort([1,4,1,2,7,5,2]));