**Ques**. Climbing Stairs

Source Code:

public class ClimbingStairs {

public int climbStairs(int n) {

if(n==1) return 1;

if(n==2) return 2;

int[] a = new int[n];

a[0]=1;

a[1]=2;

for(int i=2;i<n;i++){

a[i]=a[i-1]+a[i-2];

}

return a[n-1];

}

}

**Ques**. Subset Sum Problem

Source Code:

class Solution{

static Boolean isSubsetSum(int N, int arr[], int sum){

boolean t[][] = new boolean[N + 1][sum + 1];

for (int i = 0; i <= N; i++) {

for (int j = 0; j <= sum; j++) {

if (j==0) {

t[i][j] = true; // True

}

else if(i==0) {

t[i][j] = false;//False

}

}

}

// DP

for (int i = 1; i <= N; i++) {

for (int j = 1; j <= sum; j++) {

if (arr[i-1] <= j) {

t[i][j] = t[i-1][j-(arr[i-1])] || t[i-1][j];

} else {

// not included

t[i][j] = t[i-1][j];

}

}

}

return t[N][sum];

}

}