Given an integer array (of length n), find and return all the subsets of input array.

**Subsets are of length varying from 0 to n, that contain elements of the array. But the order of elements should remain same as in the input array.**

**Note : The order of subsets are not important.**

Input format :

Line 1 : Size of array

Line 2 : Array elements (separated by space)

#include<cmath>

using namespace std;

int subset(int input[], int n, int output[][20]) {

int b[n],i,p=0;

for(i=0;i<=pow(2,n)-1;i++)

{

int j;

for(j=0;j<n;j++)

{

b[j]=0;

}

int k,l,temp,sum;

k=n-1;

l=0;

temp=i;

int count=0;

while(temp!=0)

{

b[k--]=temp%2;

temp=temp/2;

}

for(j=0;j<n;j++)

{

if(b[j]==1)

{

count++;

}

}

output[p][l++]=count;

for(j=0;j<n;j++)

{

if(b[j]==1)

{

output[p][l++]=input[j];

// cout<<input[j]<<" ";

}

}

// cout<<endl;

p=p+1;

}

return pow(2,n);

}