Subset sum

Given a list(Arr) of N integers, print sums of all subsets in it. Output should be printed in increasing order of sums.

Example 1:

Input:
N = 2
Arr = [2, 3]
Output:
0 2 3 5
Explanation:
When no elements are taken then Sum = 0.
When only 2 is taken then Sum = 2.
When only 3 is taken then Sum = 3.
When element 2 and 3 are taken then
Sum = 2+3 = 5.

Example 2:

Input:
N = 3
Arr = [5, 2, 1]
Output:
0 1 2 3 5 6 7 8

Expected Time Complexity: O(2^N)

Expected Auxiliary Space: O(N)

Constraints:

1 <= N <= 15

0 <= Arr[i] <= 10000

```
#include<bits/stdc++.h>
#include <iostream>
using namespace std;
void solve(int a[] , int start , int end , int sum , vector<int>&v)
{
    if(start > end)
    {
        v.push_back(sum);
        return;
    }
    solve(a , start + 1 , end , sum , v);
    solve(a , start + 1 , end , sum + a[start] , v);
}
int main()
{
    int t;
    cin >> t;
    while(t--)
    {
        int n;
        cin >> n;
        int a[n];
        vector<int>v;
        for(int i = 0;i < n;i++)
            cin >> a[i];
            int start = 0 ;
            int end = n-1;
            int sum = 0;
        solve(a , start , end , sum , v);
        sort(v.begin() , v.end());
        for(int i = 0 ; i < v.size() ; i++)
           cout << v[i] << " ";
        cout << endl;
   }
        return 0;
}
```

Questions for Practice:

1) Print all subsets:

Input : abc

Output : epsilon c b bc a ac ab abc

2) Print all unique subsets :

Input : aab

Output : Null b aa aab

3) Permutations with case change :

Input : abc

Output : abc abC aBc aBC Abc AbC ABc ABC

4) Permutations with letter change :

Input : a1B2

Output : a1B2 a1b2 A1B2 A1b2

5) Print n bit binary no. having more 1 than 0 :

Given a positive integer n, print all n-bit binary numbers having more 1's than 0's for any prefix of the number.

Input : n = 2

Output : 11 10

Input : n = 4

Output : 1111 1110 1101 1100 1011 1010