

```

1  /*** IUT 2017 - Paying Bills
2  You are given N coins of value C1, C2, ... , CN and was told to pay some pending bills on M
3  different stores. Bill of i-th store is Bi. You have to find out the minimum number of coins
4  needed to pay all the bills.
5
6  There will be T ( $\leq 1000$ ) test cases. Each case starts with a positive integer N ( $\leq 15$ ) denoting the
7  number of coins. Then comes the value of each coin Ci ( $1 \leq C_i \leq 1000$ ). Next, there will be an
8  integer M ( $\leq N$ ) denoting the number of stores followed by the bill of each store Bi ( $1 \leq B_i \leq 1000$ ).
9
10 Print case number and the minimum number of coins needed. Print -1 if it is not possible to
11 pay all the bills with given coins.
12 Sample Input          Sample Output
13 2                      Case 1: 3
14 6                      Case 2: -1
15 1 2 3 4 5 6
16
17 2
18 9 3
19 6
20 1 2 3 4 5 6
21 3
22 4 4 4
23 22 // Sample Code(TIME-1.3S):
24 int n,m,coin[20],bill[20],dp[16][(1<<15)+5]; vector<int>taka[1009];
25 int cnt(int mask){
26     int one=0;
27     for(int i=0; i<n; i++){
28         if(mask&(1<<i)) one++;
29     }
30     return one;
31 }
32 int take(int mask){
33     int sum=0;
34     for(int i=0; i<n; i++){
35         if(mask & (1<<i)) sum += coin[i];
36     }
37     return sum;
38 }
39 int fun(int shop,int mask){
40     if(shop==m) return cnt(mask);
41     int &ret=dp[shop][mask];
42     if(ret!=-1) return ret;
43     ret=n+12;
44     int sz=taka[bill[shop]].size();
45     for(int i=0; i<sz; i++) {
46         int need = taka[bill[shop]][i];
47         if((mask & need)==0) {
48             ret=min(ret,fun(shop+1, mask|need));
49         }
50     }
51     return ret;
52 }
53 int main(){
54     int t; scanf("%d", &t);
55     for(int ks=1; ks<=t; ks++){
56         scanf("%d", &n); for(int i=0; i<n; i++) scanf("%d", &coin[i]);
57         scanf("%d", &m); for(int i=0; i<m; i++) scanf("%d", &bill[i]);
58
59         for(int i=1; i<(1<<n); i++){
60             int ta = take(i);
61             if(ta<=1000) taka[ta].push_back(i);
62         }
63         memset(dp,-1,sizeof(dp));
64         int ans=fun(0,0);
65         if(ans>n) ans=-1;
66         printf("Case %d: %d\n",ks,ans);
67         for(int i=0; i<1003; i++) taka[i].clear();
68     }
}

```