思路：我们知道排序后sum[1]==num[1]+num[2];sum[2]==num[1]+num[3];但是num[2]+num[3]的值是不确定的，因此我们需要i(3,m)枚举，然后求出num[3]之后，将

num[1]+num[2],num[1]+num[3],num[2]+num[3]的值标记，对于要求的num[4],在sum[]中的一个没有被标记过的一定是num[1]+num[4]的值，求出num[4]之后，在对num[1]+num[4],num[2]+num[4]….进行标记（如果都已标记，说明不成立，退出，再次枚举num[2]+num[3])；此后就可以依此求出num[5]…num[n]…如果最后成功，那么直接退出最外层循环，输出num[1]~num[n]

#include<iostream>

#include<cstdio>

#include<cstring>

#include<cmath>

#include<algorithm>

using namespace std;

const int MAXN=110;

bool mark[MAXN\*MAXN];

int sum[MAXN\*MAXN];

int num[MAXN];

/\*

设所求的n个数按从小到大排列为a1,a2....an。

a1+a2一定是n\*(n-1)/2个数的序列中最小的，a1+a3一定是次小的，通过枚举a2+a3的值解出符合条件的a1,a2,a3,

把他们两两相加的结果从n\*(n-1)/2个数中排除，然后n\*(n-1)/2个数中剩下的第一个没有被排除的数一定是a1+a4的值，

这样可以求的a4，然后再从n\*(n-1)/2个数中除去a1+a4，a2+a4，a3+a4，然后n\*(n-1)/2个数中剩下的第一个没有被排除

的数一定是a1+a5的值，这样可以求的a5，如此重复，直到求的an的值，并除去a1+an,a2+an,....an-1+an，此时如果n\*(n-1)/2

个数已经全部除去，那么就得到了答案，否则，说明枚举的a2+a3的值仍不合要求，继续枚举下一个a2+a3的值。

\*/

int main(){

int n;

while(~scanf("%d",&n)&&n){

int m=n\*(n-1)/2;

memset(sum,0,sizeof(sum));

memset(num,0,sizeof(num));

for(int i=1;i<=m;i++)scanf("%d",&sum[i]);

sort(sum+1,sum+m+1);

for(int i=3;i<=m;i++){

//枚举求出num[1],num[2],num[3];

//num[1]+num[2]一定为sum[1];

//num[1]+num[3]一定为sum[2];

//而num[2]+num[3]的值不确定，因此枚举i(3,m);

num[2]=(sum[1]-sum[2]+sum[i])/2;

num[1]=sum[1]-num[2];

num[3]=sum[2]-num[1];

if(num[2]+num[3]!=sum[i])continue;

memset(mark,false,sizeof(mark));

mark[i]=true;

int k=3;//从第三个开始排除

bool flag=true;

for(int j=4;j<=n&&flag;j++){

while(mark[k])k++;

num[j]=sum[k]-num[1];

mark[k]=true;

//去掉num[2]+num[j],num[3]+num[j],num[4]+num[j]....num[j-1]+num[j];

//那么剩下的第一个没有被排除的数一定是num[1]+num[j+1](此时是下一个循环）

for(int l=2;l<j&&flag;l++){

//排除num[j]+num[l](2<=l<=j-1)

for(int x=k+1;x<=m;x++){

flag=false;

if(!mark[x]&&num[l]+num[j]==sum[x]){

mark[x]=true;

flag=true;

break;

}

}

}

}

if(flag)break;//说明此时已经生出n个数

}

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

printf("%d ",num[i]);

}

printf("%d\n",num[n]);

}

return 0;

}