题意：从两个集合中选取元素，问最少一共选多少个元素，使得两个集合中选出的元素之和相等

解题思路：背包

对两个集合分别进行背包，dp[i]表示和为i要选出的最少元素

枚举集合元素，由于是01背包，第二层要倒序枚举i

易得dp[i]=min(dp[i],dp[i-a[j]]+1)

最后再从1一直枚举到两个集合中的元素总和的较小值，有ans=min(ans,dp1[i]+dp2[i])

代码：

#include<cstdio>

#include<algorithm>

#include<cstring>

using namespace std;

const int N=105,MAX=1e6+1;

int a[N],b[N],dpa[MAX],dpb[MAX];

int main()

{

freopen("packs.in","r",stdin);

freopen("packs.out","w",stdout);

memset(dpa,0x3f,sizeof(dpa));

memset(dpb,0x3f,sizeof(dpb));//记得初始化为正无穷

int n,m,sum=0,mmin;

scanf("%d",&n);

for(int i=1;i<=n;i++) scanf("%d",&a[i]),sum+=a[i];//求和

dpa[0]=dpb[0]=0;

for(int j=1;j<=n;j++)

for(int i=sum;i-a[j]>=0;i--)//01背包倒序枚举

dpa[i]=min(dpa[i],dpa[i-a[j]]+1);

mmin=sum;

sum=0;

scanf("%d",&m);

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

for(int j=1;j<=m;j++)

for(int i=sum;i-b[j]>=0;i--)

dpb[i]=min(dpb[i],dpb[i-b[j]]+1);

mmin=min(mmin,sum);

int ans=1e9;

for(int i=1;i<=mmin;i++) ans=min(ans,dpa[i]+dpb[i]);

if(ans<1e9) printf("%d",ans);

else puts("impossible");

return 0;

}