Design and implement in C to find a subset of a given set S = {Sl, S2,.....,Sn} of *n* positive integers whose SUM is equal to a given positive integer *d*. For example, if S ={1, 2, 5, 6, 8} and *d*= 9, there are two solutions {1,2,6}and {1,8}. Display a suitable message, if the given problem instance doesn't have a solution.

```
#include<stdio.h>
int subset(int n,int d,int w[]); int main()
{
int n,d,i,w[10];
printf("Enter the no. of elements : "); scanf("%d",&n);
printf("Enter the elements\n"); for(i=1;i<=n;i++)
scanf("%d",&w[i]);
printf("Enter the sum value : "); scanf("%d",&d);
subset(n,d,w);
return 0;
}

int subset(int n,int d,int w[])
{
int s,k,i,x[10]; for(i=1;i<=n;i++)
x[i]=0;
s=0; k=1; x[k]=1;
while(1)
{
if(k<=n && x[k]==1)
{
if(s+w[k]==d)
{
printf("Solution is \n");
for(i=1;i<=n;i++)
{
if(x[i]==1)
printf("%d ",w[i]);
}
printf("\n");
x[k]=0;
}
else if(s+w[k]<d)
    s+=w[k];
 else
 x[k]=0;
}
else
{
k--;
while(k>0 && x[k]==0)
{
k--;
}
if(k==0)
break;
x[k]=0;
s=s-w[k];
}
k=k+1;
```

```
x[k]=1;
}
return 0 ;
}
```