Design and Analysis of Algorithms – 20ISL57A

Program 11 - Implement sum of subset problem using backtracking.

```
#include<stdio.h>
#define TRUE 1
#define FALSE 0
int inc[50],w[50],sum,n;
int promising(int i,int wt,int total)
{
	return(((wt+total)>=sum)&&((wt==sum)||(wt+w[i+1]<=sum)));
}
void sumset(int i,int wt,int total)
{
	int j;
	if(promising(i,wt,total))
	{
		if(wt==sum)
		{
			printf("\n{\t");
			for (j=0;j<=i;j++)
			if(inc[j])
				printf("%d\t",w[j]);
			printf("}\n");
		}
		else
		{
			inc[i+1]=TRUE;
			sumset(i+1,wt+w[i+1],total-w[i+1]);
			inc[i+1]=FALSE;
			sumset(i+1,wt,total-w[i+1]);
		}
	}
}
```

```
int main()
{
        int i, j, n, temp, total=0;
        printf("\n Enter how many numbers:\n");
        scanf("%d",&n);
        printf("\n Enter %d numbers to the set:\n",n);
        for (i=0;i<n;i++)
        {
                scanf("%d",&w[i]);
                total+=w[i];
        }
        printf("\n Input the sum value to create sub set:\n");
        scanf("%d",&sum);
        for (i=0;i<=n;i++)
                for (j=0;j<n-1;j++)
                if(w[j]>w[j+1])
                {
                        temp=w[j];
                        w[j]=w[j+1];
                        w[j+1]=temp;
                }
        printf("\n The given %d numbers in ascending order:\n",n);
        for (i=0;i<n;i++)
           printf("%d \t",w[i]);
        if((total<sum))
           printf("\n Subset construction is not possible");
        else
        {
           for (i=0;i<n;i++)
              inc[i]=0;
           printf("\n The solution using backtracking is:\n");
           sumset(-1,0,total);
        }
}
```