**PROGRAM 17: “Sum of Subsets” problem**

Implement “Sum of Subsets” using Backtracking. “Sum of Subsets” problem: Find a subset of a given set S ={s1,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}. A suitable message is to be displayed if the given problem instance doesn’t have a solution.

**Program:**

#include <stdio.h>

#include <stdlib.h>

static int total\_nodes;

void printSubset(int A[], int size)

{

printf("{");

for(int i = 0; i < size; i++)

{

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

}

printf(" }\n");

}

void subset\_sum(int s[], int t[], int s\_size, int t\_size, int sum, int ite, int const target\_sum)

{

total\_nodes++;

if( target\_sum == sum )

{

printSubset(t, t\_size);

if( ite + 1 < s\_size && sum - s[ite] + s[ite+1] <= target\_sum )

{

subset\_sum(s, t, s\_size, t\_size-1, sum - s[ite], ite + 1, target\_sum);

}

return;

}

else

{

if( ite < s\_size && sum + s[ite] <= target\_sum )

{

for( int i = ite; i < s\_size; i++ )

{

t[t\_size] = s[i];

if( sum + s[i] <= target\_sum )

{

subset\_sum(s, t, s\_size, t\_size + 1, sum + s[i], i + 1, target\_sum);

}

}

}

}

}

void bsort(int s[],int size)

{

int i,j,temp;

for (i = 0; i < size-1; i++)

{

for (j = 0; j < size-i-1; j++)

{

if (s[j] > s[j+1])

{

temp=s[j];

s[j]=s[j+1];

s[j+1]=temp;

}

}

}

}

void generateSubsets(int s[], int size, int target\_sum)

{

int \*tuplet\_vector = (int \*)malloc(size \* sizeof(int));

int total = 0;

int i;

bsort(s, size);

for( int i = 0; i < size; i++ )

{

total += s[i];

}

if( s[0] <= target\_sum && total >= target\_sum )

{

subset\_sum(s, tuplet\_vector, size, 0, 0, 0, target\_sum);

}

free(tuplet\_vector);

}

int main()

{ int i,n;

int sets[10] ;

int target ;

printf("Enter number of elements in array\n");

scanf("%d",&n);

printf("Enter elements of sets\n");

for(i=0;i<n;i++)

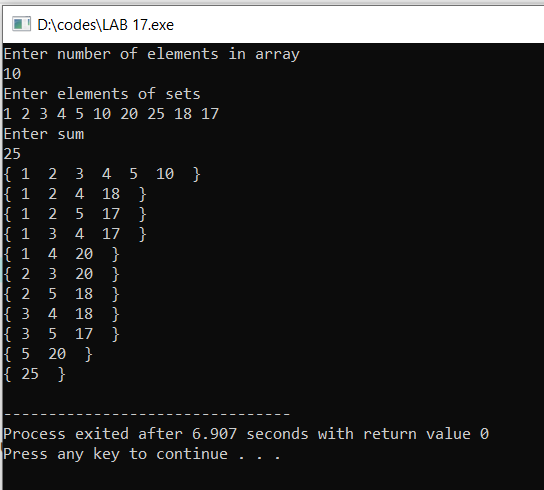
scanf("%d",&sets[i]);

printf("Enter sum\n");

scanf("%d",&target);

generateSubsets(sets,n, target);

}

****