PRACTICAL-1.4

AIM:

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 CODE:

```
#include <iostream>
using namespace std;
static int counter = 0;
bool isSubsetSum(int set[], int n, int sum)
{   counter++;
    if (sum == 0)
    return true;
    if (n == 0 && sum != 0)
        return false;
    if (set[n -1] > sum)
         return isSubsetSum(set, n -1, sum);
    return isSubsetSum(set, n -1, sum) || isSubsetSum(set, n -
1, sum -set[n -1]);
    }

int main()
{
    int n;
    int sum;
    cout<<"Enter the size of array : ";
    cin>>n;
```

```
    int set[n];

    cout<<"Enter the elements : ";
    for(int i=0;i<n;i++)
    cin>>set[i];

    cout<<"Enter the sum : ";
    cin>>sum;

    if (isSubsetSum(set, n, sum) == true)
        cout <<"SUBSET EXISTS"<<endl;
    else
        cout << "NO SUBSET EXISTS"<<endl;


        cout <<"Counter : " << counter<<endl;
        cout <<endl;
        cout<<"PARTH PATEL 19DCS098"<<endl;
        return 0;
}
```

OUTPUT:

```
Enter the size of array : 5
Enter the elements : 1 2 5 6 8
Enter the sum : 9
SUBSET EXISTS
Counter : 25

PARTH PATEL 19DCS098
```

TABLE:

N	COUNTER
2	7
3	13
4	16
5	29
6	62
7	97
8	122
9	130

GRAPH: