Practical-2.2

Student Name: Pranjal Kumar
UID: 20BCS3504
Branch: CSE
Section/Group: 607 B
Semester: 5th
Date of Performance: 14/10/2022
Subject Name: Design & Analysis Algorithm
Subject Code: 20CSP-312

1. Aim:

To implement subset-sum problem.

2. Task to be done:

Cycle through all subsets of n numbers and, for every one of them, check if the subset sum to the right number. The running time is of order O(2n.n) since there are 2n subsets, and to check each subset, we need to sum at most n elements.

3. Algorithm:

1) Include the current item in the subset and recur for the remaining items with the remaining total.

2) Exclude the current item from the subset and recur for the remaining items.

3) Finally, return true if we get a subset by including or excluding the current item; otherwise, return false.

4) Return true when the sum becomes 0, i.e., the subset is found.

4. Code:

```
#include<iostream>
using namespace std;
bool sumSet(int*arr,int sum,int n)
{
if (sum==0)
{
   return true;
}
if (n==0 && sum!=0){
   return false;
}
{
return sumSet(arr,sum,n-1);}
return sumSet(arr,sum,n- 1)||sumSet(arr,sum-arr[n-1],n-1);}
int main (){
int size;
cout<<"Enter The Size Of The Array=";
cin>>size;
int arr[size];
cout<<"Enter The Elements In The Array=";
for(int i=0;i<size;i++){
```

```
    cin>>arr[i];
}
int sum;
cout<<"Enter Targeted Sum=";
cin>>sum;
if(sumSet(arr,sum,size)==true)
{
    cout<<"Sum Found."<<endl;
}
else{
cout<<"Sum Not Found."<<endl;}
return 0;
getchar();
}
```

5. Complexity Analysis:

- **Time Complexity:** O(N*sum).
- **Auxiliary Space:** O(N*sum).

6. Result:

input

```
Enter The Size Of The Array=5
Enter The Elements In The Array=6
4
3
6
5
Enter Targeted Sum=10
Sum Not Found.


...Program finished with exit code 0
Press ENTER to exit console.
```

Learning outcomes (What I have learnt):

1. Learn about subset-sum problem.
2. Learn about time complexity of the subset-sum problem.
3. Learn about recursion.
4. Learn about initiating the best case of the condition.