WORKSHEET 6

Student Name: Harsh Kumar **UID: 22BCS15754**

Branch: CSE **Section/Group: FL_IOT_603'B'**

Semester: 5th **Date of Performance:05/09/24**

Subject Name: Design and Analysis of Algorithms **Subject Code: 22CSH-311**

1. **Aim**: Develop a program and analyze complexity to implement subset-sum problem using DynamicProgramming.

2. **Objectives:** To implement subset-sum problem using Dynamic programming.

3. **Algorithm:**

- Create a 2D array of size (n + 1) * (sum + 1) of type boolean.
- The state dp[i][j] will be true if there exists a subset of elements from set[0 . . . i] with sum value = 'j'.
- If the current element has a value greater than the 'current sum value' we will copy the answer for previous cases.
- If the current sum value is greater than the 'ith' element we will see if any of the previous states have already experienced the sum= j OR any previous states experienced a value 'j – set[i]' which will solve our purpose.

4. **Implementation/Code:**

```
#include <iostream>
#include <vector>
using namespace std;


bool isSubsetSum(int set[], int n, int sum)
{
   bool subset[n + 1][sum + 1];
```

```
 for (int i = 0; i <= n; i++)
       subset[i][0] = true;

      for (int i = 1; i <= sum; i++)
       subset[0][i] = false;

      for (int i = 1; i <= n; i++) {
       for (int j = 1; j <= sum; j++) {
          if (j < set[i - 1])
             subset[i][j] = subset[i - 1][j];
          if (j >= set[i - 1])
             subset[i][j]
                = subset[i - 1][j]
                  || subset[i - 1][j - set[i - 1]];
       }
    }

    return subset[n][sum];
}


int main()
{
    int set[] = { 3, 34, 4, 12, 5, 2 };
    int sum = 9;
    int n = sizeof(set) / sizeof(set[0]);
    if (isSubsetSum(set, n, sum) == true)
       cout << "Found a subset with given sum";
    else
       cout << "No subset with given sum";
    return 0;
}
```

5. Output:

```
Found a subset with given sum

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

6. Time Complexity:

O(sum * n), where n is the size of the array.

7. Learning Outcome:

1) Learnt how to use Dynamic Programming concepts and how to apply them to solve problems.
2) Learnt The Subset Sum Problem and how to determine if a subset with a given sum exists within a set.