**Experiment 3.2**

**Student Name:   Gaurav Kumar                                UID: 22MCC20177**

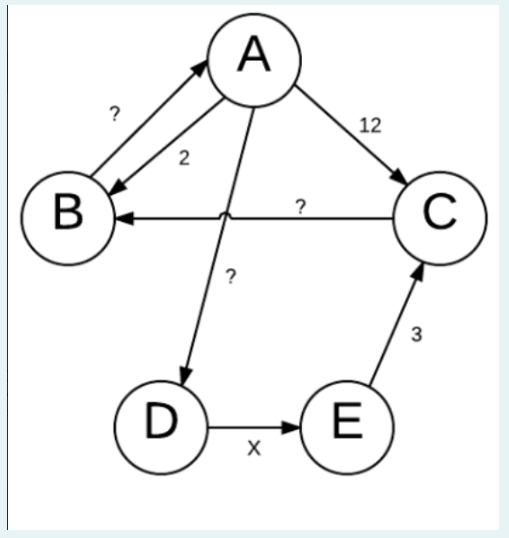
**Branch:   CC-DevOps                                                     Section/Group:- 1/B**

**Semester:   One                                                               Date of Performance: 02/01/2023**

**Subject Name:- Design & Analysis of Algorithms Lab                   Subject Code: 22CAP-646**

1. **Task to be done:**

**Find a subset of a given set S={sl,s2,sn} of n positive integers whose sum is equal to a given positive integer .For example, if S={1,2,3,4,}and d = 7 there are two solutions{1,2,4}and{3,4}. A suitable message is to be displayed if the given problem instance doesn't have a solution.**



1. **Steps for experiment/practical: copy and paste your code here/screenshots**

#include <bits/stdc++.h>

using namespace std;

bool \*\*dp;

void display(const vector<int> &v)

{

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

        printf("%d ", v[i]);

    printf("\n");

}

void printSubsetsRec(int arr[], int i, int sum, vector<int> &p)

{

    if (i == 0 && sum != 0 && dp[0][sum])

    {

        p.push\_back(arr[i]);

        if (arr[i] == sum)

            display(p);

        return;

    }

    if (i == 0 && sum == 0)

    {

        display(p);

        return;

    }

    if (dp[i - 1][sum])

    {

        vector<int> b = p;

        printSubsetsRec(arr, i - 1, sum, b);

    }

    if (sum >= arr[i] && dp[i - 1][sum - arr[i]])

    {

        p.push\_back(arr[i]);

        printSubsetsRec(arr, i - 1, sum - arr[i], p);

    }

}

void printAllSubsets(int arr[], int n, int sum)

{

    if (n == 0 || sum < 0)

        return;

    dp = new bool \*[n];

    for (int i = 0; i < n; ++i)

    {

        dp[i] = new bool[sum + 1];

        dp[i][0] = true;

    }

    if (arr[0] <= sum)

        dp[0][arr[0]] = true;

    for (int i = 1; i < n; ++i)

        for (int j = 0; j < sum + 1; ++j)

            dp[i][j] = (arr[i] <= j) ? dp[i - 1][j] || dp[i - 1][j - arr[i]]

                                     : dp[i - 1][j];

    if (dp[n - 1][sum] == false)

    {

        printf("There are no subsets with sum %d\n", sum);

        return;

    }

    vector<int> p;

    printSubsetsRec(arr, n - 1, sum, p);

}

int main()

{

    int n, sum;

    cout << "Enter the number of elements: ";

    cin >> n;

    int arr[n];

    cout << "Enter the elements: ";

    for (int i = 0; i < n; i++)

        cin >> arr[i];

    cout << "Enter the sum: ";

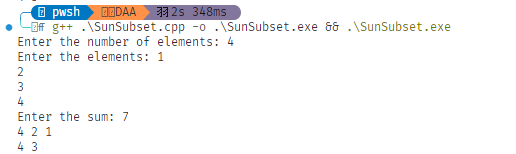
    cin >> sum;

    printAllSubsets(arr, n, sum);

    return 0;

}

1. **Output (screenshots)**

****

**Evaluation Grid:**

|  |  |  |  |
| --- | --- | --- | --- |
| Sr. No. | Parameters | Marks Obtained | Maximum Marks |
| 1. | Demonstration and Performance  (Quiz) |  | 22 |
| 2. | Worksheet |  | 8 |