CSE221 Lab Assignment 08

Task-01

In this task, I follow the Kruskal algorithm. So, at first, I sort the weighted path in an ascending order based on weight. Then I create a nested set where every vertex is individually present in a set. Additionally, I write a find function which helps to find a set of elements to find. Then I run a for loop on the path and every time try to find out if the vertices are present in the same set or not. If not, then create a new set and remove the old one and append the new one into the set. I also append the path which can be taken in the result. Finally, I get the desired answer by summing all the weight of the path from the result.

Task-02

In this task, I follow the dynamic programming's Fibonacci concept where I also use memoization. Firstly, I take an array which length is the given number times. Then I write a function named ways where I check if the given number is equal to 0 or 1, if yes, then return 1. Also, I check the number's location in the array to see if the location is empty or not. If yes, then I store the value in the location which I get from the recursive call. Finally, I return the array with the number I pass in the function and get the desired answer.

Task-03

In this task, I write a function which helps me to find the minimum number required to get the target amount. Firstly, I create an array "m" of "n" length. Then run nested loop where I check if the coin in "j" number index is less than "i" index and the min1 is greater than 1+m[i-c[j]. If it became true, then update min1 and then update m array with min1. Finally, return the array m and check if the amount number index in the array is infinity or not. If not, then output the minimum number required to get the target amount, otherwise, print –1.