Task 1:

In the first task of DSU, I write a union function that unites two different sets. For this I use the find function to find the parent of each node, and then merge them. Finally the setsize function returns the size of each set.

Task 2:

Here we use krwskal's algorithm for mst. first to sort the edges based on their cost then union them given they don't create a cycle. The union function is optimized by rank

Task 3:

counting possible ways to climb stairs. is basically Fibonacci series. we keep an array to store the value on the first visit. So that when

there is a repitition, we can just pull it from the array.

Task 4:

First I take an array till the target value. Then a loop runs to note how many steps are needed to reach each value. Calling the [target value] index of the array will give the answer.