Day 34

Code

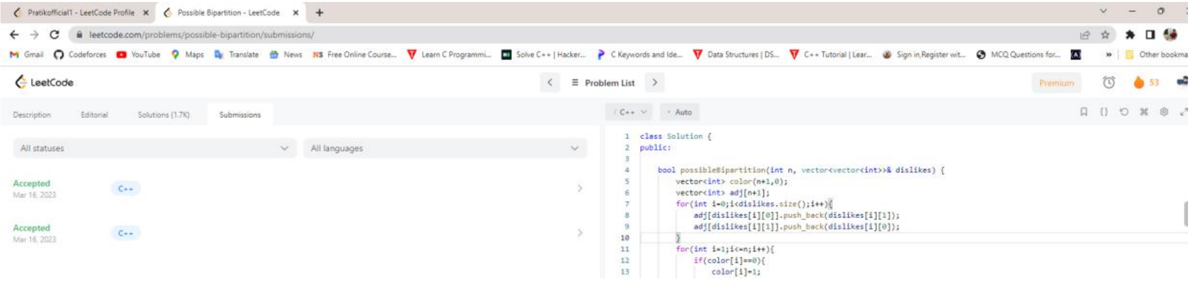
```
class Solution {
public:

    bool possibleBipartition(int n, vector<vector<int>>& dislikes) {
        vector<int> color(n+1,0);
        vector<int> adj[n+1];
        for(int i=0;i<dislikes.size();i++){
            adj[dislikes[i][0]].push_back(dislikes[i][1]);
            adj[dislikes[i][1]].push_back(dislikes[i][0]);
        }
        for(int i=1;i<=n;i++){
            if(color[i]==0){
                color[i]=1;
                queue<int> q;
                q.push(i);
                while(!q.empty()){
                    int node=q.front();
                    q.pop();
                    for(int child:adj[node]){
                        if(color[child]==color[node])return false;
                        if(!color[child]){
                            q.push(child);
                            color[child]=-1*color[node];
                        }
                    }
                }
            }
        }

        return true;
    }
};
```

Time complexity –O(N)

Space complexity –O(N)



Accepted Mar 16, 2023	C++
Accepted Mar 16, 2023	C++

```
class Solution {
public:

    bool possibleBipartition(int n, vector<vector<int>>& dislikes) {
        vector<int> color(n+1,0);
        vector<int> adj[n+1];
        for(int i=0;i<dislikes.size();i++){
            adj[dislikes[i][0]].push_back(dislikes[i][1]);
            adj[dislikes[i][1]].push_back(dislikes[i][0]);
        }
        for(int i=1;i<=n;i++){
            if(color[i]==0){
                color[i]=1;
```