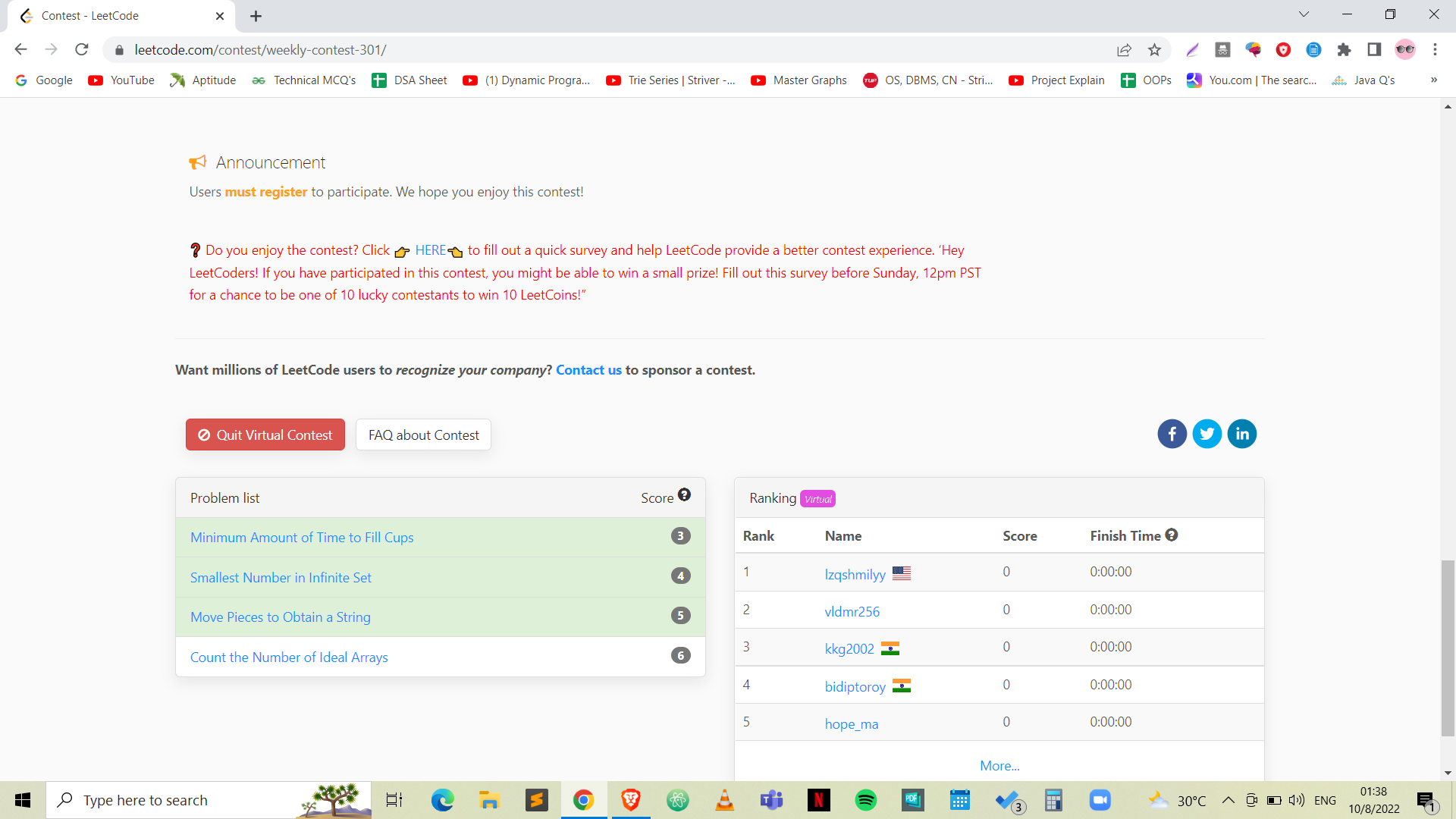
Name: Sagar Srivastava

Roll No: 1900290100127

**LeetCode Weekly Contest 301**

**Week 1**

In this contest, I solved 3/4 questions and got a rank of 3433 / 23566



The questions that I’ve solved are below:

1. Minimum Amount of Time to Fill Cups
2. Smallest Number in Infinite Set
3. Move Pieces to obtain a string
4. **Minimum Amount of Time to Fill Cups**

class Solution {

public:

int fillCups(vector<int>& amount) {

priority\_queue<int> pq;

for(auto x : amount){

if(x != 0){

pq.push(x);

}

}

int ans = 0;

while(pq.size() > 0){

if(pq.size() == 1){

int x = pq.top();

pq.pop();

ans += x;

}

else{

int a = pq.top();

pq.pop();

int b = pq.top();

pq.pop();

a--;

b--;

if(a > 0){

pq.push(a);

}

if(b > 0){

pq.push(b);

}

ans++;

}

}

return ans;

}

};

1. **Smallest Number in Infinite Set**

class SmallestInfiniteSet {

public:

set<int> s;

SmallestInfiniteSet() {

s.clear();

for(int i=1;i<=1000;i++){

s.insert(i);

}

}

int popSmallest() {

int smallest = \*s.begin();

s.erase(smallest);

return smallest;

}

void addBack(int num) {

s.insert(num);

}

};

1. **Move Pieces to Obtain a String**

class Solution {

public:

bool canChange(string start, string target) {

int n = target.length();

int m = start.length();

int a = 0, b = 0, i = 0, j = 0;

while(i < n and j < n){

while(i < n and target[i] == '\_') i++;

while(j < n and start[j] == '\_') j++;

if(i == n || j == n){

return i == n and j == n;

}

if(target[i] != start[j]) return false;

if(target[i] == 'L'){

if(j < i) return false;

}

else{

if(i < j) return false;

}

i++;

j++;

}

while(i < n and target[i] == '\_') i++;

while(j < n and start[j] == '\_') j++;

return i == n and j == n;

}

};