## Name- Akash UID- 22BCS17064

# 1763. Longest Nice Substring

class Solution { public:

string longestNiceSubstring(string s) { unordered\_set<char> missing;

for (char c : s) {

if (islower(c)) missing.insert(toupper(c)); else missing.insert(tolower(c));

}

for (int i = 0; i < s.size(); i++) { if (missing.count(s[i])) {

continue;

}

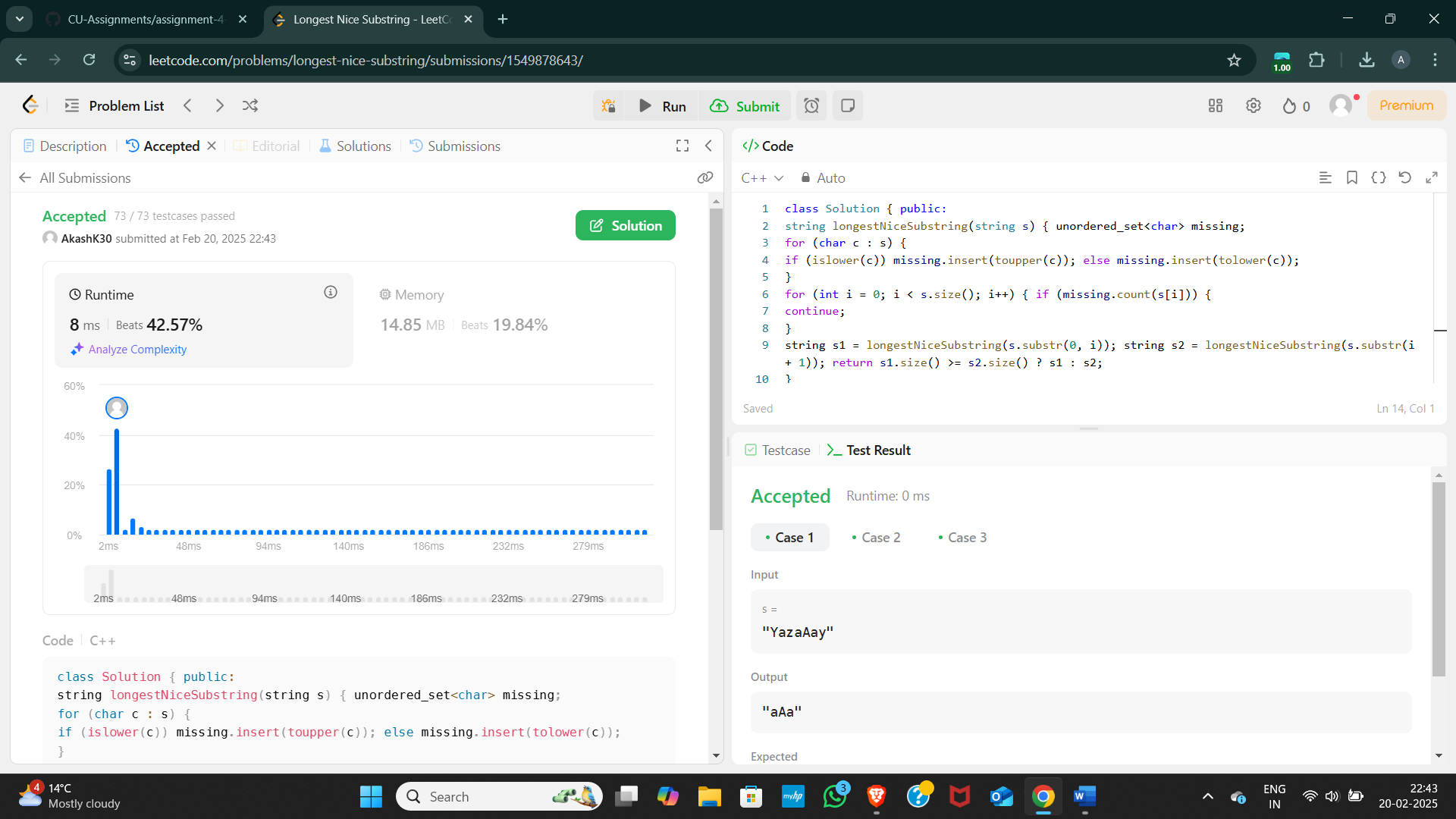
string s1 = longestNiceSubstring(s.substr(0, i)); string s2 = longestNiceSubstring(s.substr(i + 1)); return s1.size() >= s2.size() ? s1 : s2;

}

return s;

}

};



# Reverse Bits

class Solution { public:

uint32\_t reverseBits(uint32\_t n)

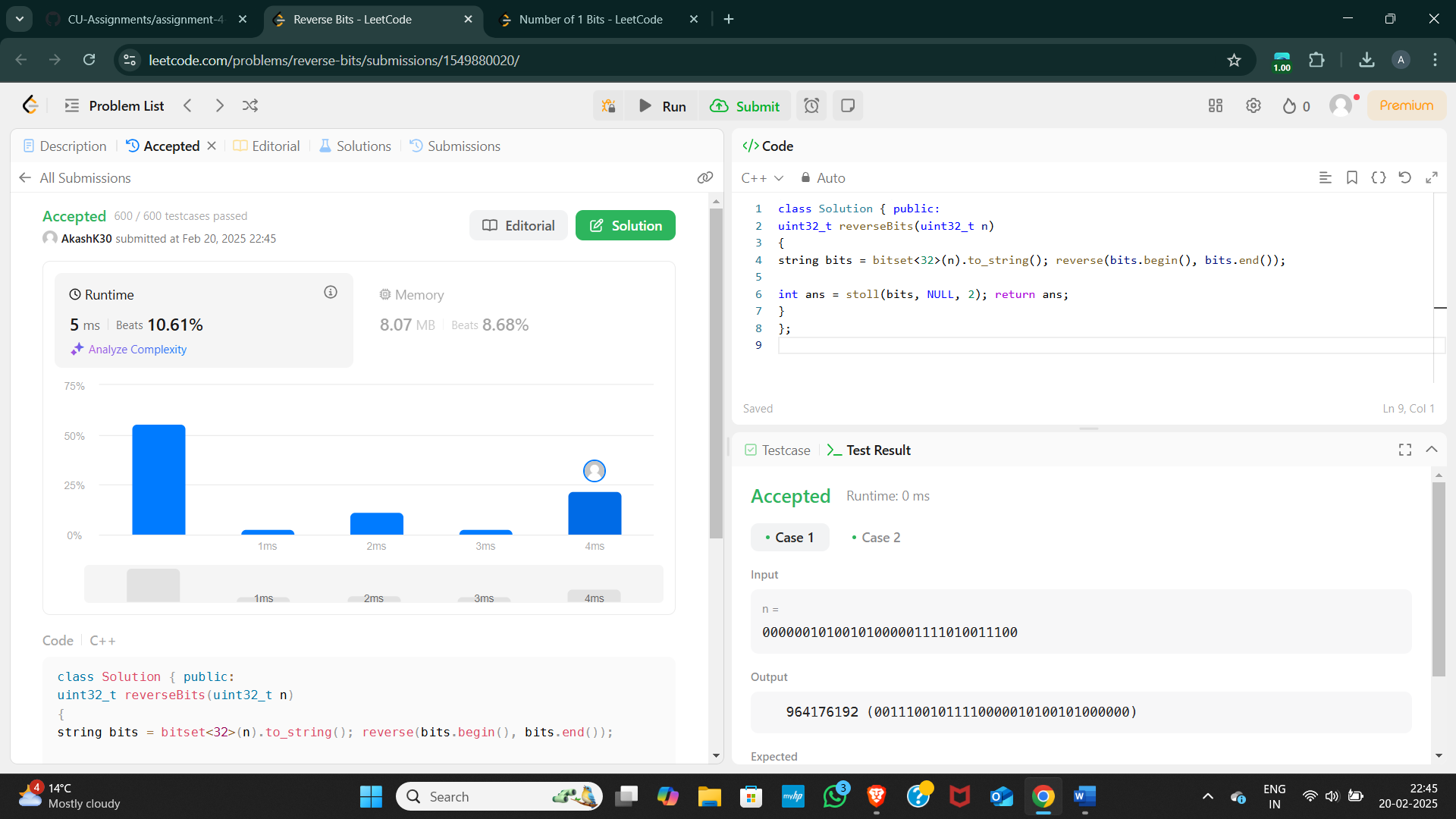
{

string bits = bitset<32>(n).to\_string(); reverse(bits.begin(), bits.end());

int ans = stoll(bits, NULL, 2); return ans;

}

};



# Number of 1 Bits

class Solution { public:

int hammingWeight(int n) { stack<int> s; while(n){

s.push(n % 2); n = n / 2;

}

int count = 0; while(!s.empty())

{

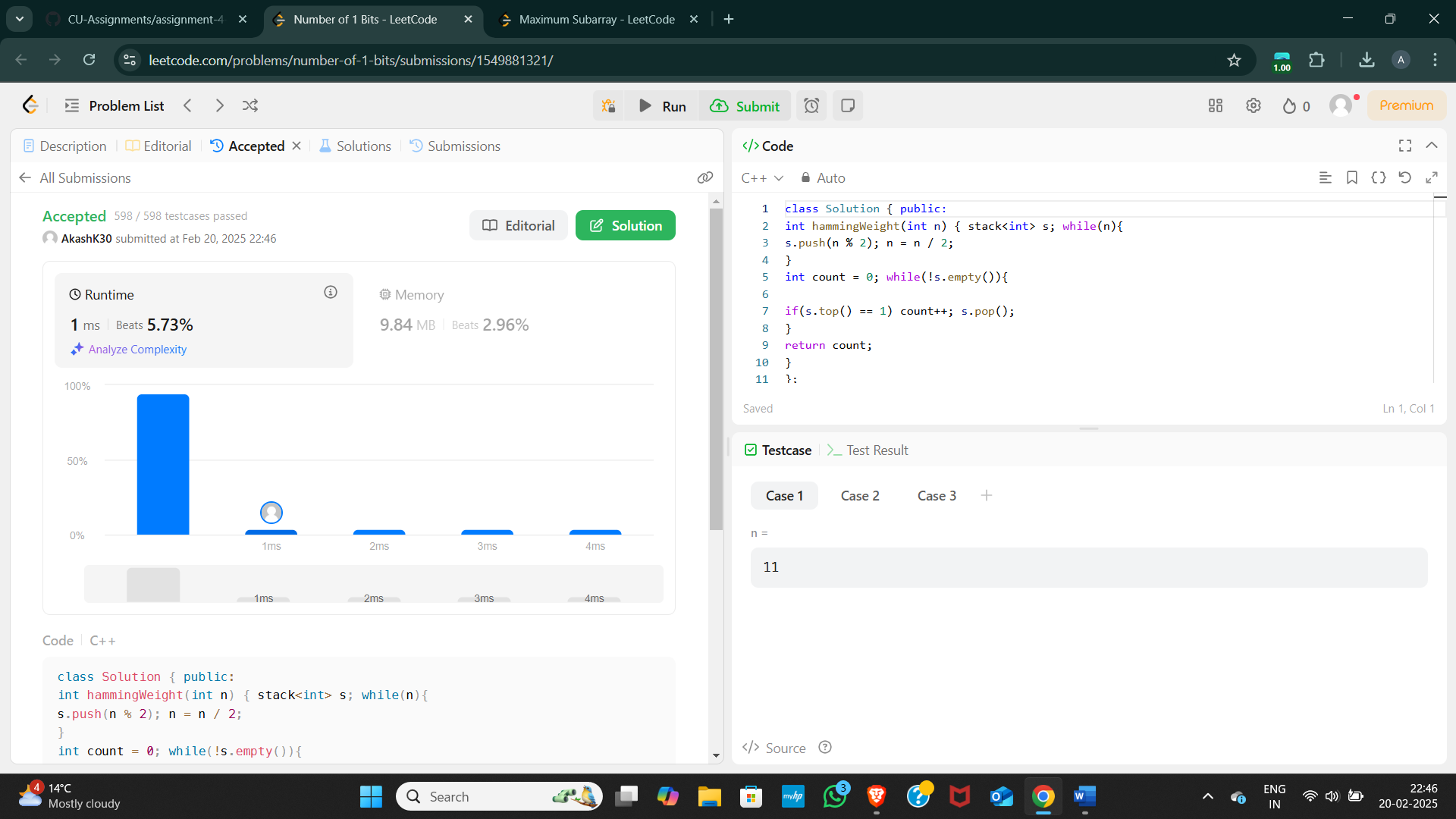
if(s.top() == 1) count++; s.pop();

}

return count;

}

};



# 53. Maximum Subarray

class Solution { public:

int maxSubArray(vector<int>& arr) {

long long maxi = LONG\_MIN; // maximum sum long long sum = 0;

int n = arr.size();

for (int i = 0; i < n; i++) {

sum += arr[i];

if (sum > maxi) { maxi = sum;

}

if (sum < 0) { sum = 0;

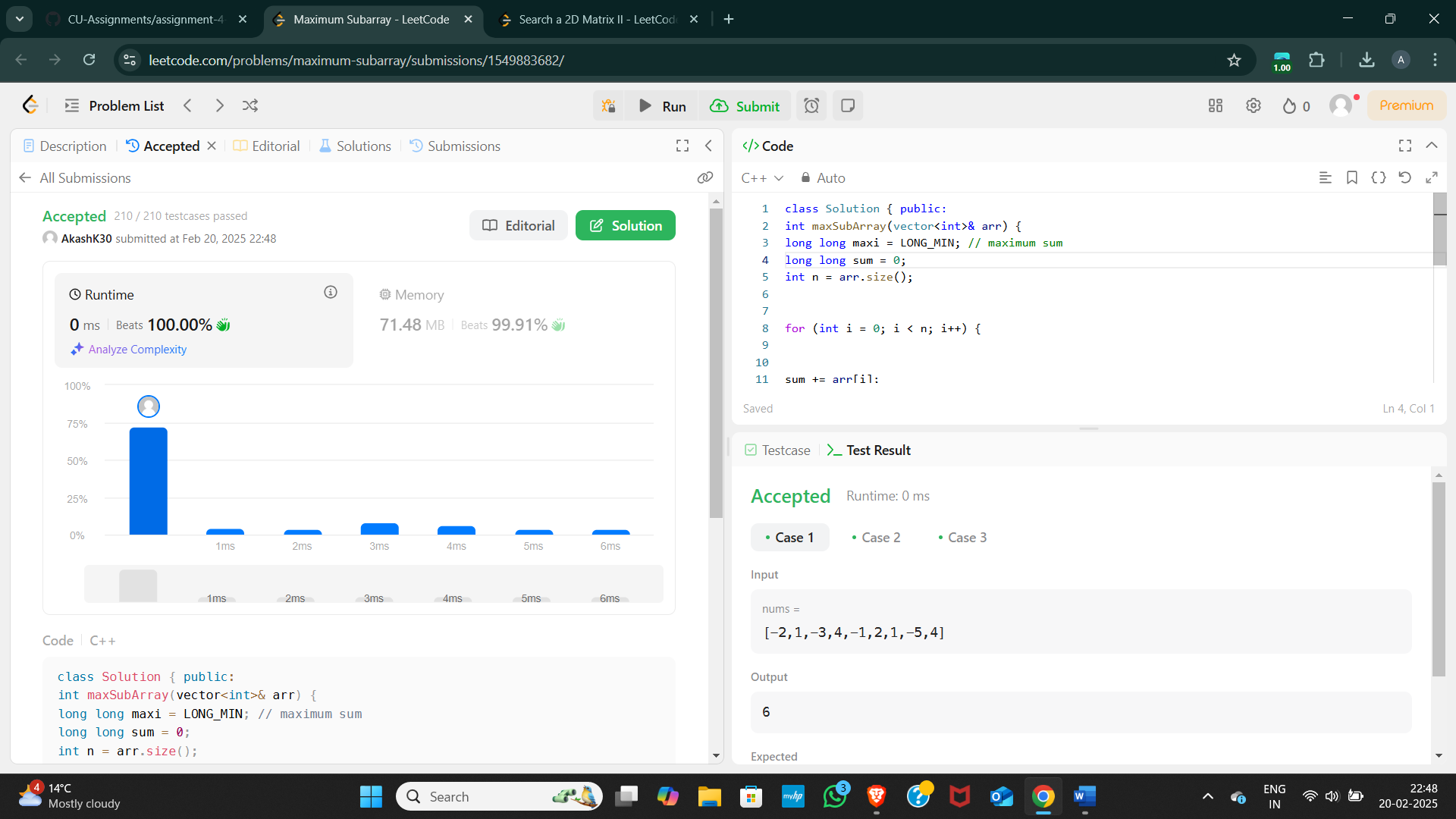
}

}

return maxi;

}

};



# 240. Search a 2D Matrix II

class Solution { public:

bool searchMatrix(vector<vector<int>>& matrix, int target) { int cols = matrix[0].size() - 1;

int n = matrix.size() - 1; int rows = 0;

while(rows <= n && cols >= 0){

int toCompare = matrix[rows][cols]; if(toCompare > target){

cols--;

}else if(toCompare < target){ rows++;

}else{

return true;

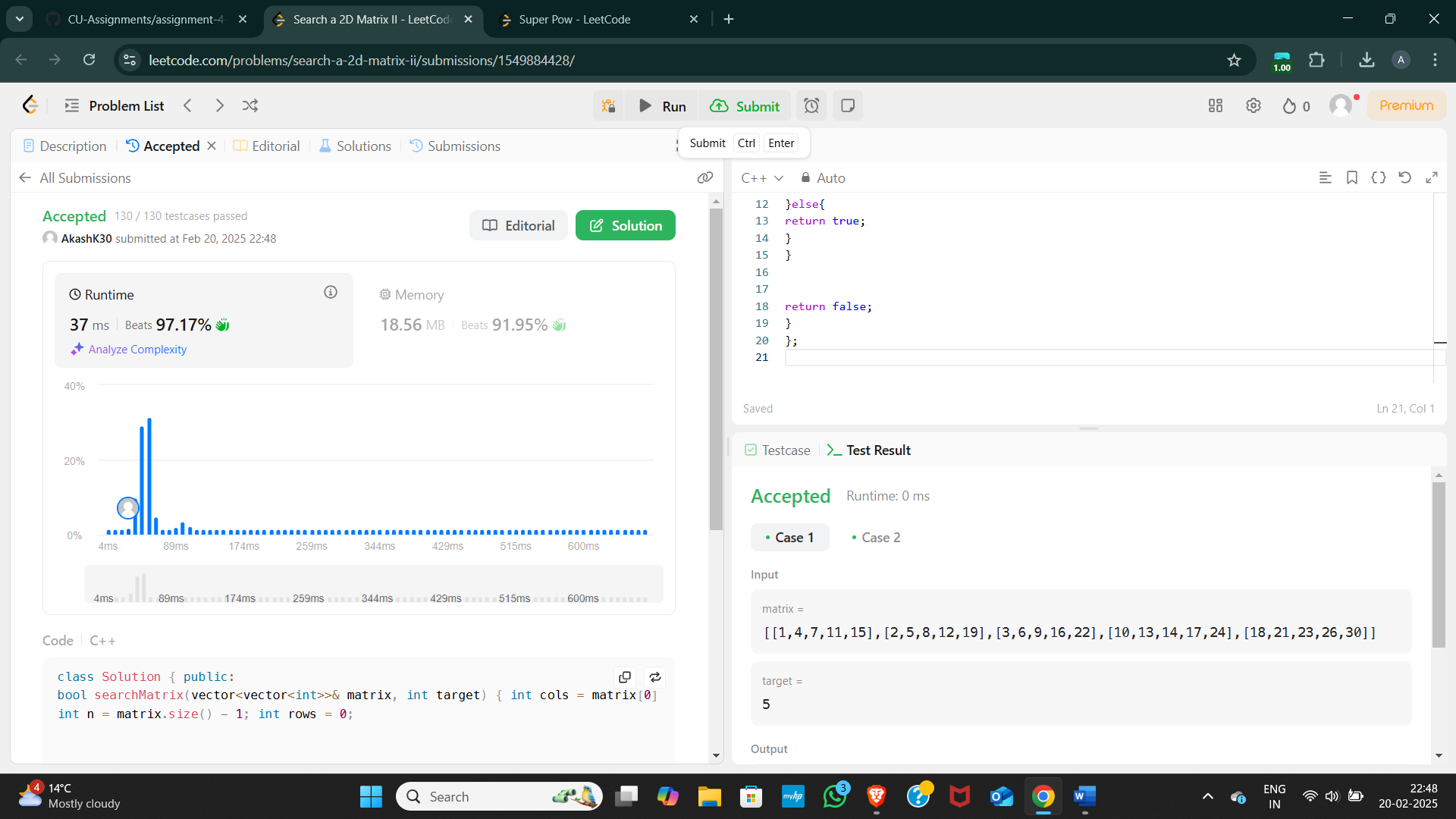
}

}

return false;

}

};



# 372. Super Pow

class Solution { public:

int pow(int a, int b){ if(b==0) return 1; int temp=pow(a,b/2);

if(b%2==0) return ((temp%1337)\*temp%1337)%1337;

else return (a%1337\*((temp%1337\*temp%1337)%1337))%1337;

}

int superPow(int a, vector<int>& b) { if(b.size()==0) return 1;

int x=b.back(); b.pop\_back();

return pow(superPow(a, b), 10) \* pow(a, x) % 1337;

}

};

