ASSIGNMENT-4(AP)

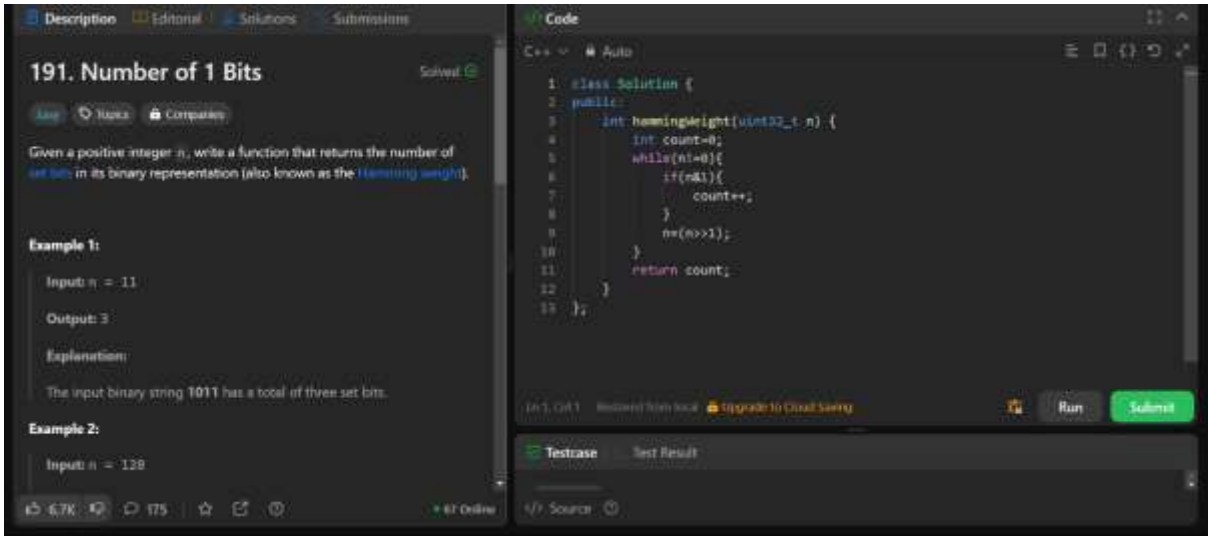
Name: Saharsh Kumar

UID: 22BCS14059

Section: 22BCS_FL_IOT-604

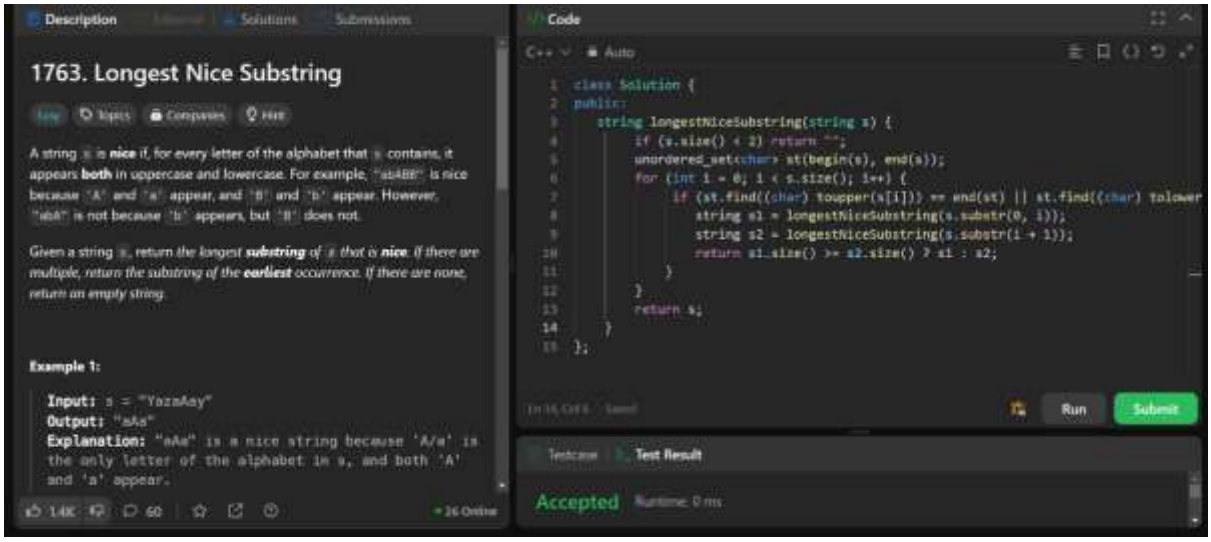
Group: A

1.) Number of 1 Bits



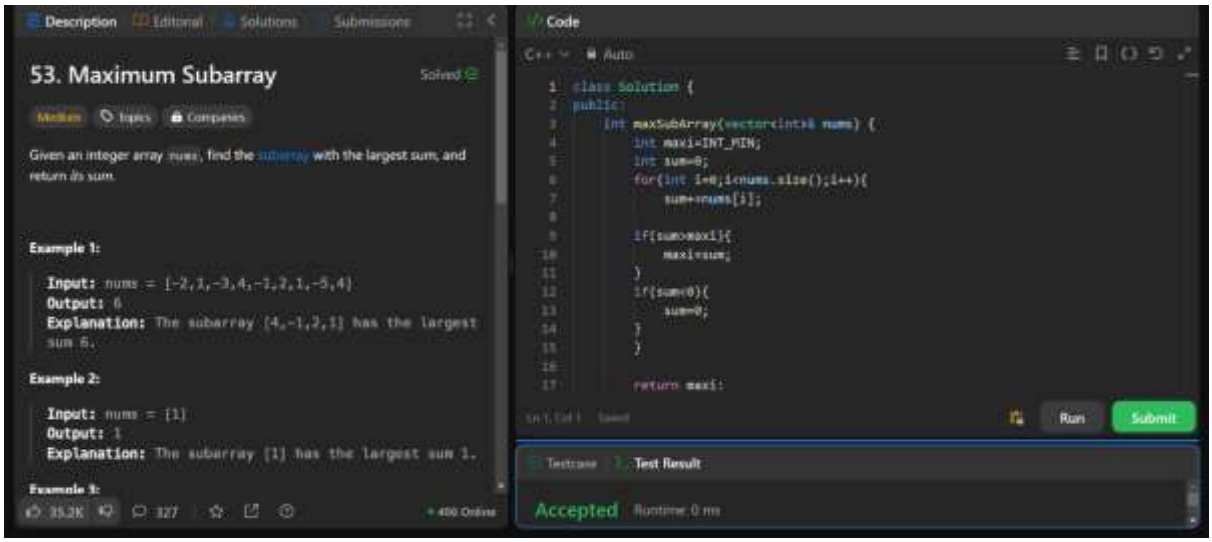

```
int hammingWeight(uint32_t n) {
    int count=0;
    while(n!=0){
        if(n&1){
            count++;
        }
        n=(n>>1);
    }
    return count;
}
```

2.) Longest Nice Substring



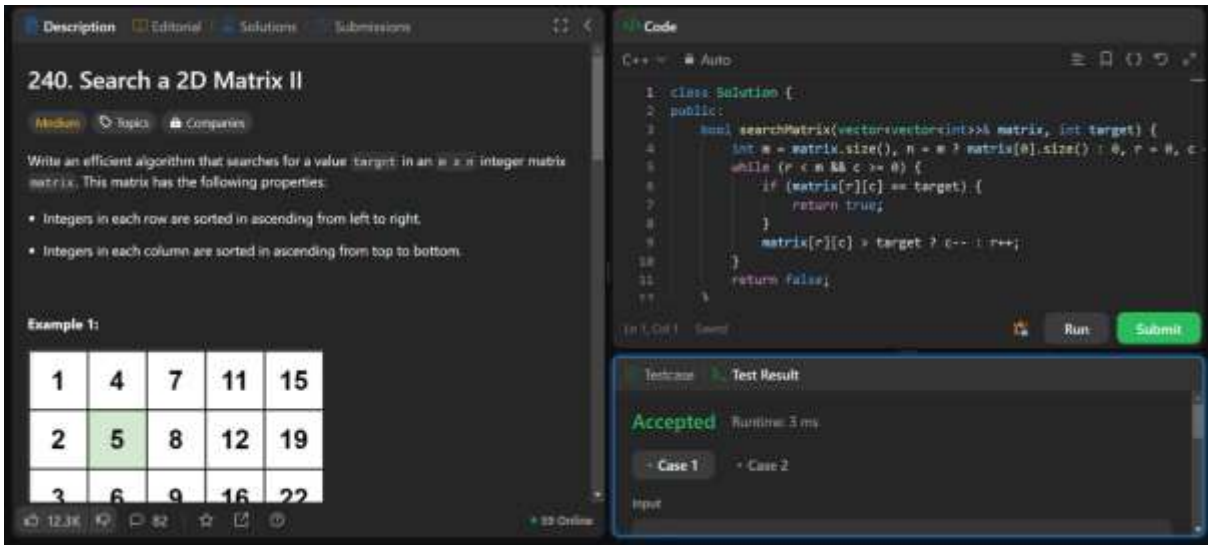

```
string longestNiceSubstring(string s) {
    if(s.size() < 2) return "";
    unordered_set<char>set(begin(s), end(s));
    for(int i = 0; i < s.size(); i++) {
        if(!set.count((char)(s[i] ^ 32))) {    //  'a' ^ 32 = 65 and 'A' ^ 32 = 97 // by XOR with 32 we can change b|w lower and uppercase
            string s1 = longestNiceSubstring(s.substr(0, i));
            string s2 = longestNiceSubstring(s.substr(i + 1));
            return s1.size() >= s2.size() ? s1 : s2;
        }
    }
    return s;
}
```

3.) Maximum Subarray



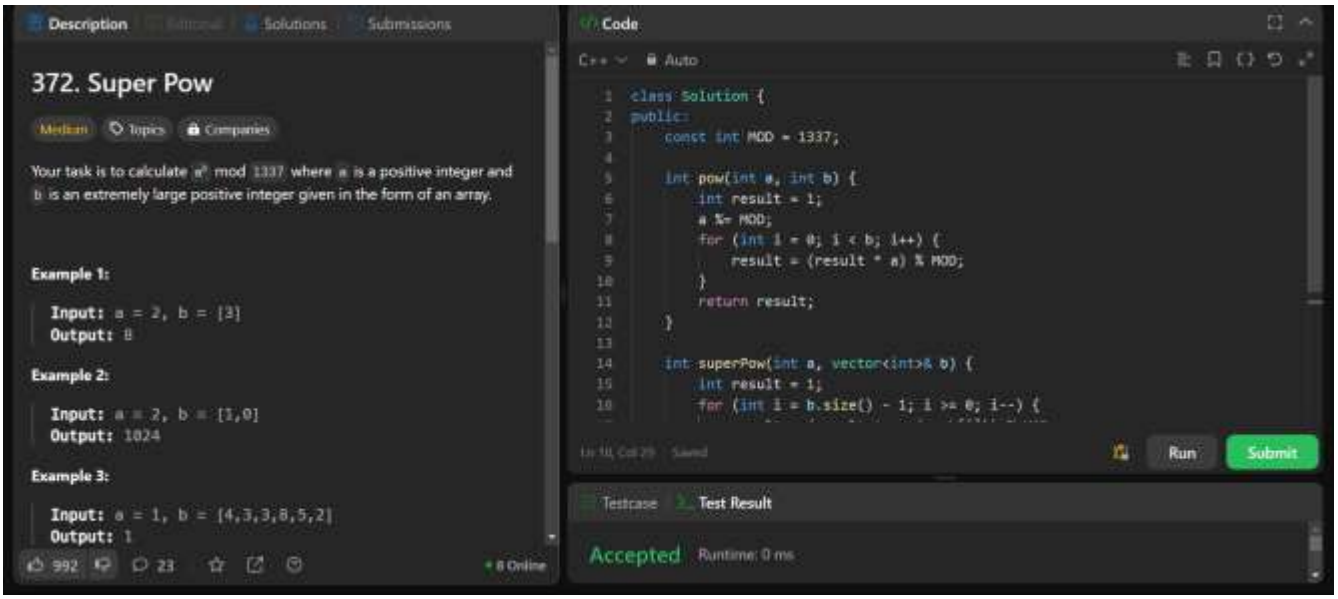

```
int maxSubArray(vector<int>& nums) {
    int maxi=INT_MIN;
    int sum=0;
    for(int i=0;i<nums.size();i++){
        sum+=nums[i];

    if(sum>maxi){
        maxi=sum;
    }
    if(sum<0){
        sum=0;
    }
    }

    return maxi;
  }
```

4.) Search a 2D Matrix II




```
bool searchMatrix(vector<vector<int>>& matrix, int target) {
    int m = matrix.size(), n = m ? matrix[0].size() : 0, r = 0, c = n - 1;
    while (r < m && c >= 0) {
        if (matrix[r][c] == target) {
            return true;
        }
        matrix[r][c] > target ? c-- : r++;
    }
    return false;
}
```

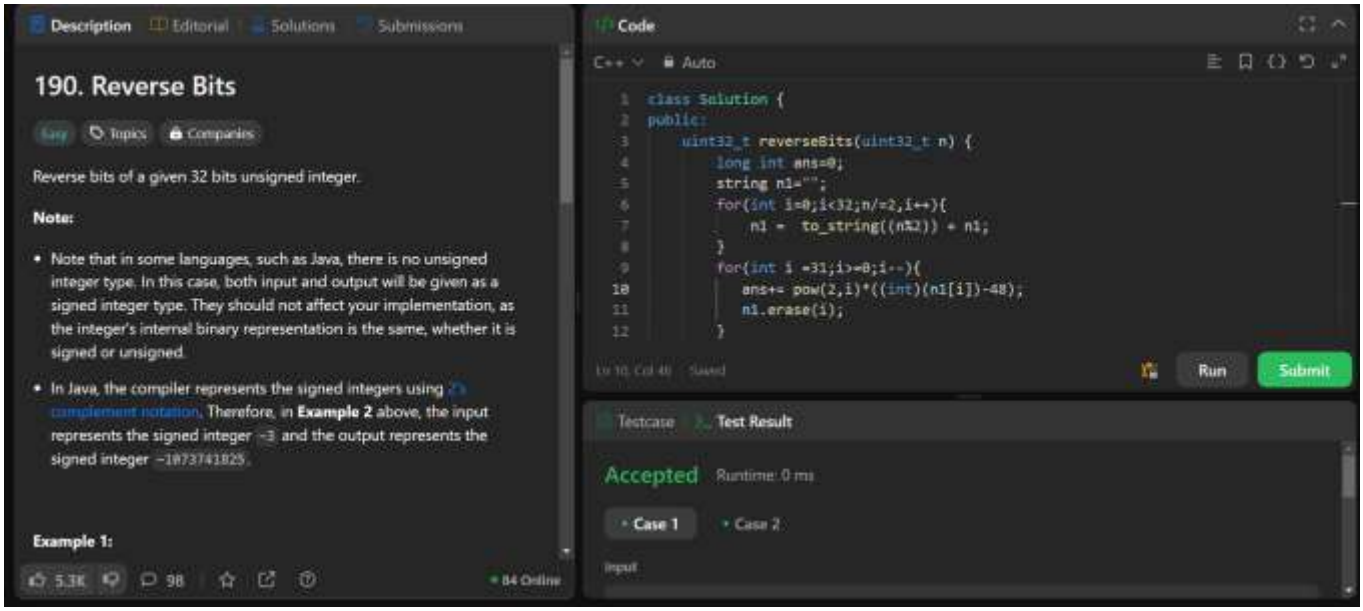
5.) Super Pow




```
const int MOD = 1337;


    int pow(int a, int b) {
        int result = 1;
        a %= MOD;
        for (int i = 0; i < b; i++) {
            result = (result * a) % MOD;
        }
        return result;
    }
    int superPow(int a, vector<int>& b) {
        int result = 1;
        for (int i = b.size() - 1; i >= 0; i--) {
            result = (result * pow(a, b[i])) % MOD;
            a = pow(a, 10);
        }
        return result;
```

}

6.) Reverse Bits




```
uint32_t reverseBits(uint32_t n) {
    long int ans=0;
    string n1="";
    for(int i=0;i<32;n/=2,i++){
      n1 =  to_string((n%2)) + n1;
    }
    for(int i =31;i>=0;i--){
      ans+= pow(2,i)*((int)(n1[i])-48);
      n1.erase(i);
    }
    return ans;
}
```