WORKSHEET 5

Student Name: Harsh Chauhan **UID:** 21BCS10053

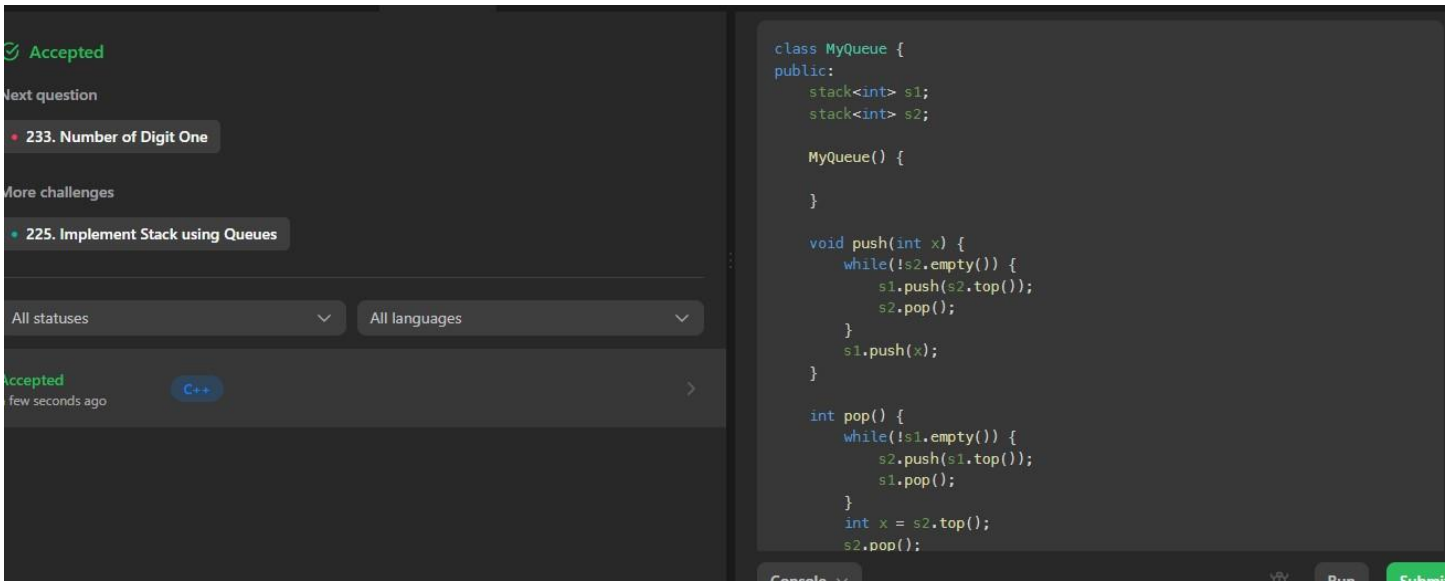
Subject Name: IT Skills (DSA)

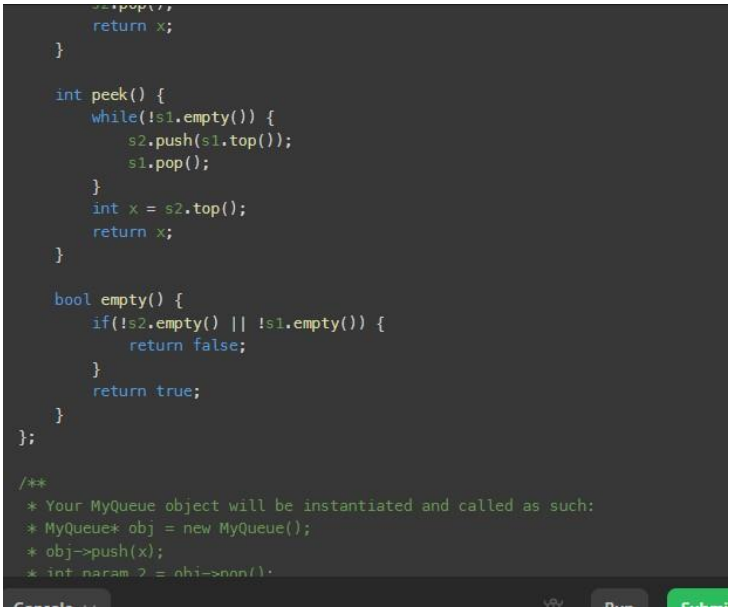
Question 1. PRODUCT OF THE LAST K NUMBERS

```
class ProductOfNumbers {
public:
    // vector 'v' will store the prefix products
    vector<int>v;
    ProductOfNumbers() {

    }

    void add(int num) {
        // if num is 0 clear the vector
        if(num==0) {
            v.clear();
            return;
        }

        // otherwise push the prefix product
        if(v.empty()) {
            v.push_back(num);
        }
        else {
            v.push_back(v.back()*num);
        }
    }

    int getProduct(int k) {
        // if k is exceeding the size of the vector that means 0 must have appeared in the stream. 0 has appeared that
        // so product must be 0
        if(k>v.size()) {
            return 0;
        }

        // otherwise find the product of the last k elements
        if(k==v.size()) {
            return v[k-1];
        }
        return v.back()/(v[v.size()-k-1]);
    }
};
```

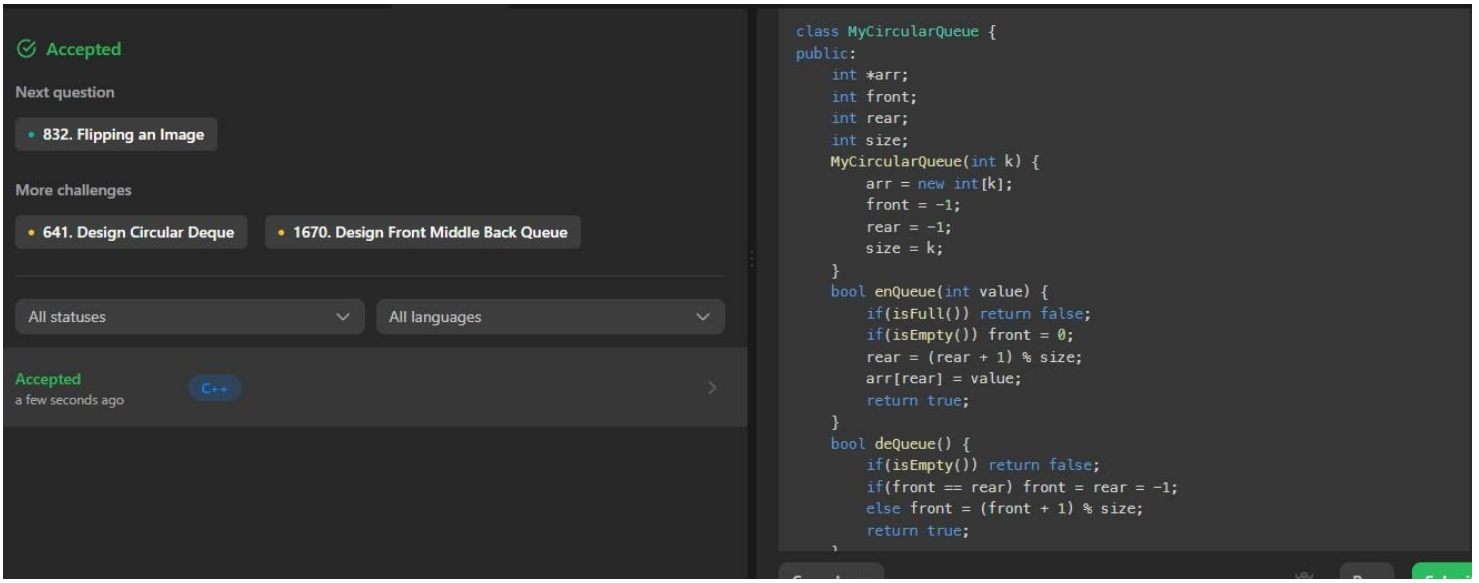
Console Run Submit

Question 2. IMPLEMENT QUEUE USING STACKS



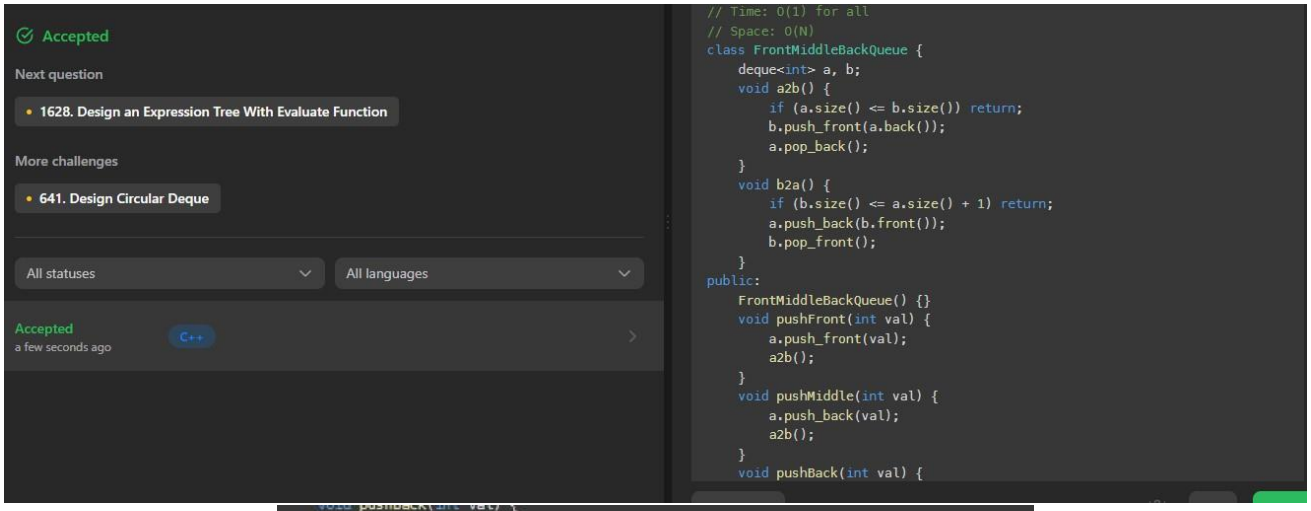


Question 3. DESIGN CIRCULAR QUEUE




```
class MyCircularQueue {
public:
    int *arr;
    int front;
    int rear;
    int size;
    MyCircularQueue(int k) {
        arr = new int[k];
        front = -1;
        rear = -1;
        size = k;
    }
    bool enQueue(int value) {
        if(isFull()) return false;
        if(isEmpty()) front = 0;
        rear = (rear + 1) % size;
        arr[rear] = value;
        return true;
    }
    bool deQueue() {
        if(isEmpty()) return false;
        if(front == rear) front = rear = -1;
        else front = (front + 1) % size;
        return true;
    }
    int Front() {
        if(isEmpty()) return -1;
        return arr[front];
    }
    int Rear() {
        if(isEmpty()) return -1;
        return arr[rear];
    }
    bool isEmpty() {
        return front == -1;
    }
    bool isFull() {
        return ((rear + 1) % size) == front;
    }
};
```

Question 4. DESIGN FRONT MIDDLE BACK QUEUE




```cpp
// Time: O(1) for all
// Space: O(N)
class FrontMiddleBackQueue {
    deque<int> a, b;
    void a2b() {
        if (a.size() <= b.size()) return;
        b.push_front(a.back());
        a.pop_back();
    }
    void b2a() {
        if (b.size() <= a.size() + 1) return;
        a.push_back(b.front());
        b.pop_front();
    }
public:
    FrontMiddleBackQueue() {}
    void pushFront(int val) {
        a.push_front(val);
        a2b();
    }
    void pushMiddle(int val) {
        a.push_back(val);
        a2b();
    }
    void pushBack(int val) {
        b.push_back(val);
        b2a();
    }
    int popFront() {
        if (a.empty() && b.empty()) return -1;
        int ans;
        if (a.empty()) {
            ans = b.front();
            b.pop_front();
        } else {
            ans = a.front();
            a.pop_front();
            b2a();
        }
        return ans;
    }
    int popMiddle() {
        if (a.empty() && b.empty()) return -1;
        int ans;
        if (a.size() == b.size()) {
            ans = a.back();
            a.pop_back();
        } else {
            ans = b.front();
            b.pop_front();
        }
        return ans;
    }
    int popBack() {
        if (a.empty() && b.empty()) return -1;
        int ans = b.back();
        b.pop_back();
        a2b();
        return ans;
    }
```

Question 5. BOOK EXERCISES

Language: C++14

```cpp
#include <iostream>
using namespace std;
#include<bits/stdc++.h>
int main() {
    // your code goes here
    ios_base::sync_with_stdio(false);
    cin.tie(NULL);
    int n;
    cin>>n;
    stack<pair<pair<int,string>,int>> st1;
    for(int i=0;i<n;i++){
        int x;
        cin>>x;
        string str;
        if(x!=-1 ){
            cin>>str;
        }else if(x==-1){
           cout<<st1.top().second<<" "<<st1.top().first.second<<endl;
            st1.pop();
            continue;
        }
        if (x==0){
            continue;
        }
        if(!st1.empty() and st1.top().first.first<x ){
           st1.top().second++;
           continue;
       }

           pair<int,string> p = make_pair(x,str);
           pair<pair<int,string>,int> whole_pair= make_pair(p,0);
           st1.push(whole_pair);
    }
    return 0;
}
```

0:0

SOLUTION:

Status: ✔ Correct Answer — Submission ID: 84843119

Time:	Memory:
0.82s	27.3M

Question 6. SUBARRAY SUM

Language: C++14

```cpp
#include <bits/stdc++.h>
using namespace std;

#define ll long long
#define rep(i,a,b) for(int i = a; i <= b; ++i)
#define repr(i,a,b) for(int i = a; i>=b; --i)
#define all(a) a.begin(), a.end()
#define pb push_back
#define N 1005
#define ii pair <int, int>
#define se second
#define fi first
#define bit(i) (i & (-i))
#define base 29
#define mod 1000000007ll

int o1[4] = { 0, 1, 0, -1 };
int o2[4] = { -1, 0, 1, 0 };

string yno[2] = { "NO\n", "YES\n" };

void print(vector <int>& a) {
    for (int i : a) cout << i << ' ';
    cout << endl;
}

void print(ii a) {
    cout << a.first << ' ' << a.second << endl;
}

void print(int a[], int n) {
    rep(i, 1, n) cout << a[i] << ' ';
    cout << endl;
}
```

```cpp
}

int p[200005];

void init() {
    rep(i, 1, 200000) {
        p[i] = p[i - 1] + i;
        p[i] %= mod;
    }
}

void solve() {
    int n;
    cin >> n;
    vector <int> a(n + 2);
    rep(i, 1, n) {
        cin >> a[i];
    }
    vector <int> l(n + 2), r(n + 2);
    stack <int> s;
    rep(i, 1, n) {
        while (s.size() && a[s.top()] < a[i]) {
            s.pop();
        }
        if (s.size()) {
            l[i] = s.top();
        }
        else l[i] = 0;
        s.push(i);
    }
    s = stack <int>();
    repr(i, n, 1) {
        while (s.size() && a[s.top()] <= a[i]) {
            s.pop();
        }
        if (s.size()) {
            r[i] = s.top();
```

```
            r[i] = s.top();
        }
        else r[i] = n + 1;
        s.push(i);
    }
    ll ans = 0;
    rep(i, 1, n) {
        ll t = (1ll * a[i] * (i - l[i])) % mod;
        t = t * ((1ll * p[r[i] - i]) % mod);
        ans += t;
        ans %= mod;
        t = (1ll * a[i] * (i - l[i])) % mod;
        t = t * ((1ll * (r[i] - i) * (n + 1 - r[i])) % mod);
        ans += t;
        ans %= mod;
    }
    cout << ans << endl;
}

int main() {
    ios_base::sync_with_stdio(false); cin.tie(0);
#ifndef ONLINE_JUDGE
#endif
    init();
    int t = 1;
    cin >> t;
    while (t--) {
        solve();
    }
    return 0;
}
```

SOLUTION:

Status: ✔ Correct Answer — Submission ID: 85012356

Score:	Time:	Memory:
100	0.11s	7.1M

Sub-Task	Task #	Result (time)
1	1	AC (0.105002)
1	2	AC (0.094270)
1	3	AC (0.097227)
1	4	AC (0.095927)
1	5	AC (0.077491)
1	6	AC (0.097477)
1	7	AC (0.094990)
1	8	AC (0.083370)
Subtask Score: 100.00%		Result - AC
		Total Score = 100.00%

Question 7. ABSOLUTE MIN MAX

Language: C++14

```cpp
#include <bits/stdc++.h>

using namespace std;
using ll=long long;

#define pb push_back

ll ans,n;
ll arr[250002];
vector<ll> li[250002];

struct fenwick
{
    std::vector<ll> tree;

    void init(ll n)
    {
        ll i;
        tree.resize(n+5);
        for(i=0;i<=n;i++)
        {
            tree[i]=0;
        }
    }

    void upd(ll i,ll v)
    {
        for(++i;i<=n;i+=(i&(-i)))
        {
            tree[i]+=v;
        }
    }

    ll calc(ll i)
    {
        ll sum=0;
        for(++i;i>0;i-=(i&(-i)))
        {
            sum+=tree[i];
        }
        return sum;
    }

    ll calc(ll l,ll r)
    {
        return calc(r)-calc(l-1);
    }
};

void solve()
{
    stack<ll> stck;
    ll i;
    ll nxt[n+5];
    fenwick tr;
    tr.init(n+5);
    for(i=0;i<=n;i++)
    {
        li[i].clear();
    }
    for(i=0;i<n;i++)
    {
        while(!stck.empty() && arr[stck.top()]<=arr[i])
        {
            stck.pop();
        }
        if(stck.empty())
        {
            tr.upd(i,1);
        }
        else
        {
            li[stck.top()].pb(i);
        }
        stck.push(i);
    }
    while(!stck.empty()){stck.pop();}
    for(i=n-1;i>=0;i--)
    {
        while(!stck.empty() && arr[stck.top()]>=arr[i])
        {
            stck.pop();
        }
        if(stck.empty())
        {
            nxt[i]=n;
        }
        else
        {
            nxt[i]=stck.top();
        }
        stck.push(i);
    }
    for(i=0;i<n;i++)
```

```
    for(i=0;i<n;i++)
    {
        for(ll j:li[i])
        {
            tr.upd(j,1);
        }
        ans+=tr.calc(i,nxt[i]-1);
    }
    return ;
}

int main()
{
    ios_base::sync_with_stdio(false);
    cin.tie(NULL);
    ll testcases, i, cur;
    cin >> testcases;
    while (testcases--)
    {
        cin >> n;
        ans = 0;
        for (i = 0; i < n; i++)
        {
            cin >> arr[i];
        }
        solve();
        reverse(arr, arr + n);
        solve();
        cur = 1;
        for (i = 1; i < n; i++)
        {
            if (arr[i - 1] == arr[i])
            {
                cur++;
            }
            else
            {
                ans -= (cur * (cur + 1)) / 2;
                cur = 1;
            }
        }
        ans -= (cur * (cur + 1)) / 2;
        cout << ans << "\n";
    }
    return 0;
}
```

SOLUTION:

Status: ✓ Correct Answer — Submission ID: 85012429

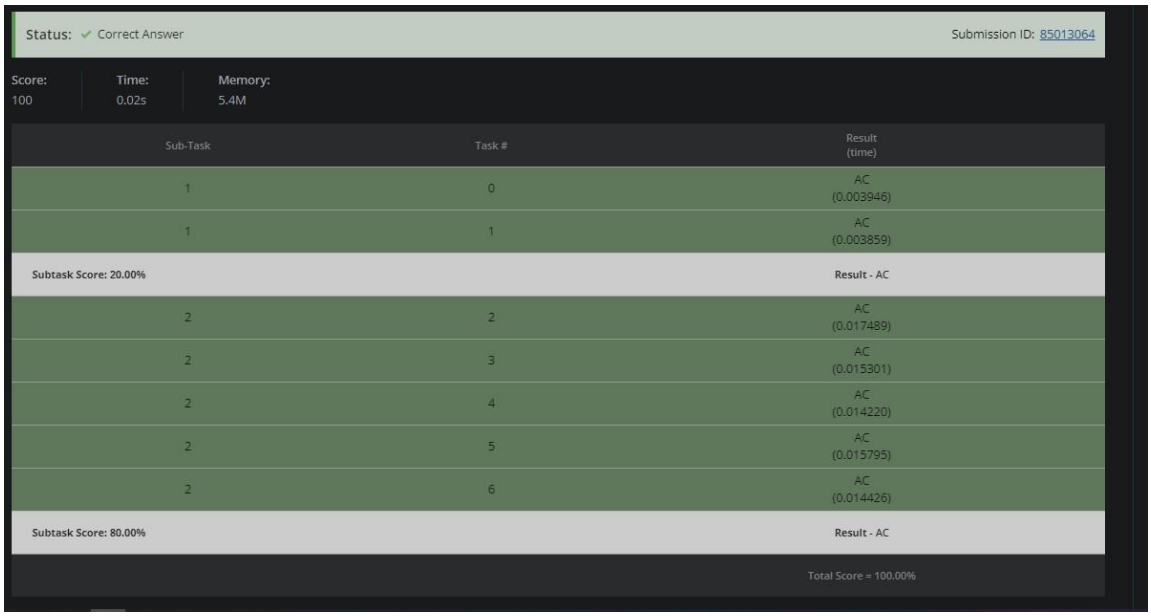
Score:	Time:	Memory:
100	0.56s	35.6M

Sub-Task	Task #	Result (time)
1	0	AC (0.005792)
1	1	AC (0.007151)
1	2	AC (0.008080)
1	3	AC (0.009422)
Subtask Score: 20.00%		Result - AC
2	4	AC (0.485408)
2	5	AC (0.418104)
2	6	AC (0.555810)
2	7	AC (0.508494)
Subtask Score: 80.00%		Result - AC
		Total Score = 100.00%

Question 8. WEAK IN THE MIDDLE

```cpp
#include <bits/stdc++.h>
using namespace std;

void solve() {
    int n;
    cin >> n;
    vector<int> t(n, 0);
    stack<tuple<int, int, int>> st;

    for(int i = 0; i < n; ++i) {
        int a, end_time = 0;
        cin >> a;

        while(st.size() >= 2) {
            auto x = st.top();
            st.pop();
            auto y = st.top();
            if(get<0>(x) < min(get<0>(y), a))
                t[get<2>(x)] = end_time = 1 + max(get<1>(x), end_time);
            else {
                st.push(x);
                break;
            }
        }

        st.push({a, end_time, i});
    }

    for(int T : t) cout << T << ' ';
    cout << '\n';
}

int main() {
    ios_base :: sync_with_stdio(false);
    cin.tie(0);
    int t;
    cin >> t;
    while(t--) solve();
    return 0;
}
```

SOLUTION:



Status: ✔ Correct Answer — Submission ID: 85013064

Score:	Time:	Memory:
100	0.02s	5.4M

Sub-Task	Task #	Result (time)
1	0	AC (0.003946)
1	1	AC (0.003859)
Subtask Score: 20.00%		**Result - AC**
2	2	AC (0.017489)
2	3	AC (0.015301)
2	4	AC (0.014220)
2	5	AC (0.015795)
2	6	AC (0.014426)
Subtask Score: 80.00%		**Result - AC**
		Total Score = 100.00%