



## Hussain Set

---

```
#include <bits/stdc++.h>
using namespace std;
typedef long long ll;
int main()
{
    ll n, m, lm, x, a, j, ptr;
    cin >> n >> m;
    vector<ll> pq;
    queue<ll> q;
    vector<ll> res;
    ll arr[m];
    for (ll i = 0; i < n; i++)
    {
        cin >> x;
        pq.push_back(x);
```

```

}

for (ll i = 0; i < m; i++)

{
    cin >> arr[i];
}

sort(pq.begin(),pq.end());

ptr=n-1;

lm = arr[m - 1];

while (lm--)

{
    if(!q.empty()){
        if(ptr>=0 && pq[ptr]>q.front()){
            a=pq[ptr];

            ptr--;

            res.push_back(a);

            a=a/2;

            if(a>0)

                q.push(a);
        }

        else{

            a=q.front();

            q.pop();

            res.push_back(a);

            a=a/2;

            if(a>0)

```

```
        q.push(a);
    }
}
else{
    a=pq[ptr];
    ptr--;
    res.push_back(a);
    a=a/2;
    if(a>0)
        q.push(a);
    }
}
for (ll i = 0; i < m; i++)
{
    cout << res[arr[i] - 1] << endl;
}
return 0;
}
```

## Iron, Magnet and Wall

---

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

```
void solve() {  
    int n,k;  
    cin >> n >> k;  
    string s;  
    cin >> s;  
    queue <int> qm, qi;  
    int pos = 1, ans = 0;  
    bool flag;  
    for(auto c:s) {  
        switch(c) {  
            case 'X':  
                while(!qm.empty()) qm.pop();  
                while(!qi.empty()) qi.pop();  
                break;  
  
            case ':':  
                pos ++;  
                break;  
  
            case 'I':  
                flag = false;  
                while(!qm.empty()) {  
                    int a = qm.front();  
                    qm.pop();  
                    if(k+1-abs(pos-a) > 0) {
```

```
        ans ++;

        flag = true;

        break;

    }

}

if(!flag) qi.push(pos);

break;

case 'M':

    flag = false;

    while(!qi.empty()) {

        int a = qi.front();

        qi.pop();

        if(k+1-abs(pos-a) > 0) {

            ans ++;

            flag = true;

            break;

        }

    }

    if(!flag) qm.push(pos);

    break;

}

pos ++;

}

cout << ans << endl;
```

```
}
```

```
int main() {  
    int t;  
    cin >> t;  
    while(t--){  
        solve();  
    }  
    return 0;  
}
```

## Dr Phil goes to the ranch

---

```
#include<bits/stdc++.h>  
  
#define ll long long  
#define fi first  
#define se second  
#define endl "\n"  
#define pb push_back  
#define DEBUG cout << "This-part-has-been-reach" << endl;  
#define DEBUGG(X) cout << X << endl;  
#define DEBUGGG(X, Y) cout << X << " " << Y << endl;  
#define fastio(); ios_base :: sync_with_stdio(false); cin.tie(NULL); cout.tie(NULL);  
#define tc int T; cin>>T; while(T--)
```

```
using namespace std;

typedef unsigned int uint;

typedef unsigned long long ull;

typedef pair<int, int> pii;

typedef pair<long long, long long> pll;

typedef vector<string> vs;

typedef vector<int> vi;

typedef vector<ll> vll;

typedef vector<pii> vpii;

typedef vector<pll> vpll;

typedef stack<int> sti;

typedef stack<ll> stll;

typedef stack<string> stt;

int main(){

    fastio();

    tc{

        stack<int> st;

        int n,c;

        cin >> n;

        vector<bool> v(n+1,false);

        int stt = 0;

        while(n--){

            cin >> c;

            if(c>stt){

                cout << c << " ";
```

```
        stt++;
        while(c>stt) st.push(stt++);
    }
    v[c] = true;
    while(!st.empty() and v[st.top()]){
        cout << st.top() << " ";
        st.pop();
    }
}
cout << endl;
}
}
```

## Corona in Karunanagar

---

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

#define ll long long

using namespace std;

int main(){
    ll t,n,m,i,j;
    cin>>t;
    while(t--){
        cin>>n;
        string s;
        cin>>s;
```



```
cin>>m;

vector<ll>v(m);

for(i=0; i<m; i++){
    cin>>v[i];
}

queue<ll>q;

for(i=0; i<n; i++){
    if(s[i]=='1'){
        q.push(i);
    }
}

ll ans=0;

map<ll, ll>mp;

for(i=0; i<m; i++){

    ll id = v[i]-1;

    mp[id]++;

    ll c = q.size();

    while(c>0){

        ll top = q.front();

        q.pop();

        if(mp[top]==0 && s[top-1] == '0' ){
```

```

        q.push(top-1);
        s[top-1]='1';
    }
    if(mp[top+1]==0 && s[top+1]=='0'){
        q.push(top+1);
        s[top+1]='1';
    }

    c--;
}
}
//cout<<s<<" < == s \n";
for(i=0; i<n; i++){
    if(s[i]=='1'){
        ans++;
    }
}
cout<<ans<<"\n";
}
}

```

## Cinema Ticketing Queue

---

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

```
using namespace std;
```

```
const int N=1e5+1;

int n,ans[N];

deque<int> d[N];

set<int> s;

void solve(){

    cin>>n;

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

        int x;

        cin>>x;

        s.insert(x);

        d[x].push_back(i);

    }

    for(int i=1;i<=2*n;i++){

        auto it=s.upper_bound(i);

        if(it==s.begin())continue;

        it--;

        int p=d[*it].front();

        d[*it].pop_front();

        ans[p]=i;

        if(d[*it].empty())s.erase(it);

    }

    for(int i=1;i<=n;i++)

        cout<<ans[i]<<' ';

    cout<<'\n';
```

```
}
```

```
int main(){  
    int t;cin>>t;  
    while(t--){  
        solve();  
    }  
    return 0;  
}
```