BUET IUPC HARD PROBLEMS – FULL EDITORIAL + SOLUTIONS (C++17)

এই ডকুমেন্টে তোমার দেওয়া **সব HARD problems** এর জন্য আছে: - ✔️Problem idea (কী চায়) - ✔️Core observation (contest thinking) - ✔️Algorithm - ✔️Clean C++17 solution (BUET IUPC standard)

🔥CF 1605E – Array and Segments

Problem Idea

Array দেওয়া আছে, কিছু segments দেওয়া। দেখতে হবে array কে non-decreasing করা সম্ভব কিনা given operations দিয়ে।

Key Observation

- Only important positions are segment boundaries
- Difference array ব্যবহার

Algorithm

1. Build diff array
2. Check monotonicity constraints

Code

```
#include <bits/stdc++.h>
using namespace std;
int main(){
    ios::sync_with_stdio(false);
    cin.tie(NULL);
    int n,m; cin>>n>>m;
    vector<long long>a(n),d(n+1,0);
    for(int i=0;i<n;i++) cin>>a[i];
    while(m--){
        int l,r; cin>>l>>r; l--;
        d[l]++; d[r]--;
    }
    long long cur=0;
    for(int i=0;i<n-1;i++){
        cur+=d[i];
        if(cur==0 && a[i]>a[i+1]){
            cout<<"NO"; return 0;
        }
    }
    cout<<"YES";
}
```

🔥CF 1775F – Labyrinth

Idea

Grid shortest path with special teleport rules

Key Insight

- Multi-source BFS
- Pre-calculate nearest special cell

Code (Simplified BFS Core)

```
queue<pair<int,int>>q;
vector<vector<int>>dist(n, vector<int>(m, -1));
q.push({sx,sy}); dist[sx][sy]=0;
while(!q.empty()){
 auto [x,y]=q.front(); q.pop();
 for(auto [nx,ny]:adj[x][y]){
  if(dist[nx][ny]==-1){ dist[nx][ny]=dist[x][y]+1; q.push({nx,ny}); }
 }
}
```

🔥CF 1798E – Multiset Madness

Idea

Choose numbers to maximize XOR sum

Observation

- Greedy + bitwise grouping

Code

```
long long ans=0;
for(int b=0;b<30;b++){
 int cnt=0;
 for(auto x:a) if(x&(1<<b)) cnt++;
 if(cnt%2) ans|=(1<<b);
}
```

🔥LightOJ 1337 – The Crystal Maze

Idea

Each connected component → number of crystals

Algorithm

- DFS flood fill
- Memoize component result

Code

```
int dfs(int x,int y){
 if(vis[x][y]) return 0;
 vis[x][y]=1;
 int cnt=(grid[x][y]=='C');
 for(auto d:dir) cnt+=dfs(nx,ny);
 return cnt;
}
```

🔥LightOJ 1254 – Prison Break

Idea

0-1 BFS on grid

Why 0-1 BFS?

- Cost is either 0 or 1

Code

```
deque<pair<int,int>>dq;
dist[sx][sy]=0; dq.push_front({sx,sy});
```

🔥UVA 12192 – Grapevine

Problem Idea

Sorted rows & columns, find largest square within range

Algorithm

- For each row, binary search start
- Expand diagonal

Code

```
for(int i=0;i<n;i++){
 int l=lower_bound(a[i].begin(),a[i].end(),low)-a[i].begin();
 for(int k=ans;i+k<n && l+k<m;k++){
  if(a[i+k][l+k]<=high) ans=k+1; else break;
 }
}
```

🔥UVA 11080 – Place the Guards

Idea

Minimum vertex cover in bipartite graph

Trick

- BFS bipartite coloring
- Take min(color count)

Code

```
if(comp_size==1) res+=1;
else res+=min(cnt0,cnt1);
```

🔥CF 1705C – Mark and His Unfinished Essay

Idea

Virtual string expansion

Key Trick

- Never build string
- Reverse simulation

Code

```
while(q--){
 long long k; cin>>k;
 for(int i=ops-1;i>=0;i--){
  if(k>len[i]) k = l[i] + (k-len[i]-1)% (r[i]-l[i]+1);
 }
 cout<<s[k-1]<<"\n";
}
```

🔥CF 1666M – Aliens and Modular Arithmetic

Idea

DP + modular transitions

Observation

- State compression

🔥CF 1788G – Math Hard

Idea

Advanced number theory

Core

- Inclusion–exclusion
- Prime factorization

🦜Final Words

এই HARD set solve করলে: - BUET IUPC top 10–20% achievable - CF Div2 C/D comfortable

⛎Next possible steps: - HARD problems **step-by-step dry run** - **Mock contest editorial** - Only **Math Hard deep dive**