Олимпиада СПбГУ
по информатике
2023/24 учебного года

24fin0201 – Кильметов Данил Булатович

A	B	C	D	E	F	sum
100	100	100	-	40	55	395

Task A

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

using namespace std;
using ll = long long;
using pii = pair<ll, ll>;
#define all(a) a.begin(), a.end()

int main() {
    #ifndef LOCAL
    ios_base::sync_with_stdio(0);
    cin.tie(0);
    cout.tie(0);
    #endif // LOCAL
    ll a, b, n;
    cin >> a >> b >> n;
    vector<ll> dp(n + 1);
    dp[0] = 1;
    for (ll i = 1; i <= n; ++i) {
        if (i >= a) dp[i] += dp[i - a];
        if (i >= b) dp[i] += dp[i - b];
    }
    ll ans = dp[n];
    if (n - a >= 0) ans += (a - 1) * dp[n - a];
    if (n - b >= 0) ans += (b - 1) * dp[n - b];
    cout << ans;
}
```

Task B

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

using namespace std;
using ll = long long;
using pii = pair<ll, ll>;
#define all(a) a.begin(), a.end()

ll get(ll x1, ll y1, ll z1, ll x2, ll y2, ll z2) {
    //cout << x1 << " " << y1 << " " << z1 << " " << (abs(x1 - x2) + abs(
    y1 - y2) + abs(z1 - z2)) / 2 << endl;
    return (abs(x1 - x2) + abs(y1 - y2) + abs(z1 - z2)) / 2;
}

int main() {
    #ifndef LOCAL
    ios_base::sync_with_stdio(0);
    cin.tie(0);
    cout.tie(0);
    #endif // LOCAL
    ll n;
    cin >> n;
    ll x1, y1, z1;
    cin >> x1 >> y1 >> z1;
    ll x2, y2, z2;
    cin >> x2 >> y2 >> z2;
    //if (x1 > x2) swap(x1, x2);
    //if (y1 > y2) swap(y1, y2);
    //if (z1 > z2) swap(z1, z2);
    ll ans = get(x1, y1, z1, x2, y2, z2);

    ans = min(ans, get(x1 + n, y1 + n, z1 - 2 * n, x2, y2, z2));
    ans = min(ans, get(x1 - n, y1 - n, z1 + 2 * n, x2, y2, z2));

    ans = min(ans, get(x1 + n, y1 - 2 * n, z1 + n, x2, y2, z2));
    ans = min(ans, get(x1 - n, y1 + 2 * n, z1 - n, x2, y2, z2));

    ans = min(ans, get(x1 - 2 * n, y1 + n, z1 + n, x2, y2, z2));
    ans = min(ans, get(x1 + 2 * n, y1 - n, z1 - n, x2, y2, z2));

    cout << ans;
}
```

Task C

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

using namespace std;
using ll = long long;
using pii = pair<ll, ll>;
#define all(a) a.begin(), a.end()

const int N = 1e5;
const int P = 3, Q = 1e9 + 7;
```

```
vector<int> p;

int add(int a, int b) {
    int x = a + b;
    if (x >= Q) x -= Q;
    return x;
}

int sub(int a, int b) {
    int x = a - b;
    if (x < 0) x += Q;
    return x;
}

int mult(int a, int b) {
    return (int)(1ll * a * b % Q);
}

vector<int> h(string &s) {
    int n = s.size();
    vector<int> hs(n + 1);
    for (int i = 1; i <= n; ++i) {
        hs[i] = add(mult(hs[i - 1], P), s[i - 1] - '0' + 1);
    }
    return hs;
}

int subh(vector<int> &hs, int l, int r) {
    return sub(hs[r], mult(hs[l], p[r - l]));
}

void first() {
    int n;
    cin >> n;
    string x;
    cin >> x;
    vector<int> hs = h(x);
    cout << hs[n];
}

void second() {
    int n;
    cin >> n;
    string x;
    cin >> x;
    int hs;
    cin >> hs;
    vector<int> ht = h(x);
    for (int i = 0; i < n; ++i) {
        int hPref = subh(ht, i, n);
        int hSuf = subh(ht, 0, i);
        int tmp = add(mult(hPref, p[i]), hSuf);
        if (tmp == hs) {
```

```
            cout << "YES";
            return;
        }
    }
    cout << "NO";
}

int main() {
    #ifndef LOCAL
    ios_base::sync_with_stdio(0);
    cin.tie(0);
    cout.tie(0);
    #endif // LOCAL
    string s;
    cin >> s;
    p.resize(N + 1);
    p[0] = 1;
    for (int i = 1; i <= N; ++i) {
        p[i] = mult(p[i - 1], P);
    }
    if (s == "first") {
        first();
    } else {
        second();
    }
}
```

Task E

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

using namespace std;
using ll = long long;
#define all(a) a.begin(), a.end()

vector<vector<int>> g;
vector<int> shift;
vector<int> order;
vector<int> cycle;
ll timer = 0;
vector<bool> used;
ll unused = 0;
bool cycleStarted = false;
ll prv = -100;
ll mx = -1;

void dfs(ll v, ll p) {
    if (!used[v]) --unused;
    used[v] = true;
    while (true) {
        //if (timer < 100) cout << v << " " << unused << endl;
        if (v == 0 && unused == 0) {
            if (cycleStarted) {
```

```
                return;
            } else {
                ll n = g.size();
                cycleStarted = true;
                used.assign(n, false);
                used[0] = 0;
                prv = p;
                unused = n - 1;
            }
        }
        if (mx <= 100000 && cycle.size() + order.size() >= 100010) return
    ;
        if (!cycleStarted) order.push_back(v);
        else cycle.push_back(v);
        if (order.size() >= 20000000) exit(-1);
        ++timer;
        ll nxt = g[v][shift[v]];
        ++shift[v];
        if (shift[v] >= g[v].size()) shift[v] = 0;
        if (nxt != p) dfs(nxt, v);
        else return;
    }
}

int main() {
    ios_base::sync_with_stdio(0);
    cin.tie(0);
    cout.tie(0);
    ll n, q;
    cin >> n >> q;
    g.resize(n);
    shift.resize(n);
    used.resize(n);
    for (ll i = 0; i < n; ++i) {
        ll k;
        cin >> k;
        g[i].resize(k);
        for (ll j = 0; j < k; ++j) {
            cin >> g[i][j];
            g[i][j]--;
        }
    }
    vector<ll> qry(q);
    for (int i = 0; i < q; ++i) {
        cin >> qry[i];
        mx = max(mx, qry[i]);
    }
    unused = n;
    dfs(0, -1);
    for (int i = 0; i < q; ++i) {
        ll t = qry[i];
        if (t < order.size()) {
            cout << order[t] + 1 << " ";
```

```
        } else {
            t -= order.size();
            cout << cycle[t % cycle.size()] + 1 << " ";
        }
    }
}
```

Task F

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

using namespace std;
using ll = long long;
using pii = pair<ll, ll>;
#define all(a) a.begin(), a.end()

mt19937 rnd(777);
ll n, m;
vector<string> a;
vector<vector<pii>> g;
vector<ll> mt, mtw, used;

bool kuhn(ll v, ll it) {
    if (used[v] == it) return false;
    used[v] = it;
    for (auto [to, w] : g[v]) {
        if (mt[to] == -1) {
            mt[to] = v;
            mtw[to] = w;
            return true;
        }
    }
    for (auto [to, w] : g[v]) {
        if (kuhn(mt[to], it)) {
            mt[to] = v;
            mtw[to] = w;
            return true;
        }
    }
    return false;
}

int main() {
    //#ifndef LOCAL
    ios_base::sync_with_stdio(0);
    cin.tie(0);
    cout.tie(0);
    //#endif // LOCAL
    cin >> n >> m;
    a.resize(n);
    g.resize(n * m);
    mt.resize(n * m, -1);
    mtw.resize(n * m, -1);
```

```
used.resize(n * m, -1);
vector<ll> mn(n * m, 100);
vector<ll> order(n * m);
for (ll i = 0; i < n; ++i) {
    cin >> a[i];
}
for (ll i = 0; i < n; ++i) {
    for (ll j = 0; j < m; ++j) {
        order[i * m + j] = i * m + j;
        ll x = a[i][j] - '0', y;
        for (ll di = -1; di <= 1; ++di) {
            for (ll dj = -1; dj <= 1; ++dj) {
                if (di != 0 && dj != 0) continue;
                if (di == 0 && dj == 0) continue;
                ll i1 = i + di, j1 = j + dj;
                if (i1 >= n) i1 = 0;
                if (i1 < 0) i1 = n - 1;
                if (j1 >= m) j1 = 0;
                if (j1 < 0) j1 = m - 1;
                y = a[i1][j1] - '0';
                g[i * m + j].push_back({i1 * m + j1, x * y});
                mn[i * m + j] = min(mn[i1 * m + j1], x * y);
            }
        }
        sort(all(g[i * m + j]), [](pii x, pii y) {
             return x.second > y.second;
        });
        //shuffle(all(g[i * m + j]), rnd);
    }
}
sort(all(order), [&mn](ll x, ll y) {
     return mn[x] > mn[y];
});
ll need = (order[0] / m) + (order[0] % m);
for (auto el : order) {
    ll x = (el / m) + (el % m);
    if (x % 2 == need % 2) {
        kuhn(el, el);
    }
}
ll ans = 0;
vector<string> res(n, string(m, '.'));
for (ll i = 0; i < n; ++i) {
    for (ll j = 0; j < m; ++j) {
        if (mt[i * m + j] != -1) {
            ans += mtw[i * m + j];
            ll x = mt[i * m + j];
            ll i1 = x / m, j1 = x % m;
            if (i == i1) {
                if ((j + 1) % m == j1) {
                    res[i][j] = 'L';
                    res[i1][j1] = 'R';
                } else {
```

```
                        res[i][j] = 'R';
                        res[i1][j1] = 'L';
                    }
                } else {
                    if ((i + 1) % n == i1) {
                        res[i][j] = 'T';
                        res[i1][j1] = 'B';
                    } else {
                        res[i][j] = 'B';
                        res[i1][j1] = 'T';
                    }
                }
            }
        }
    }
    cout << ans << "\n";
    for (auto el : res) {
        cout << el << "\n";
    }
}
```