Олимпиада СПбГУ
по информатике
2023/24 учебного года

24fin0451 – Тихонов Илья Владимирович

A	B	C	D	E	F	sum
100	100	100	35	20	49	404

Task A

```
/*
-Werror
g++ solution.cpp -o solution.exe -D_GLIBCXX_DEBUG -Wall -Wextra -Wshadow
clang++ solution.cpp -o solution.exe -fsanitize=undefined -fsanitize=
    address -g -g3 -Wall -Wextra
*/

//#ifdef LOCAL  //-DLOCAL
//# debug(x) cerr << __LINE__ << ") " << (#x) << " = " << (x) << endl
//#else
//# debug(x) ;
//#endif

//#pragma GCC optimize("Ofast")//O3
//#pragma GCC target("avx2")

#include <algorithm>
#include <cmath>
#include <fstream>
#include <iostream>
#include <map>
#include <numeric>
#include <queue>
#include <random>
#include <set>
#include <stack>
#include <string>
#include <unordered_map>
#include <unordered_set>
```

```
#include <vector>
#include <iomanip>
#include <bitset>
#include <cassert>
#include <chrono>

using namespace std;
using ld = long double;
using ll = long long;
using ull = unsigned long long;
//using BigInt = __int128_t;
mt19937 rnd(chrono::high_resolution_clock::now().time_since_epoch().count
    ());

void solve() {
    vector<vector<ll>> C(100, vector<ll>(100));
    C[0][0] = 1;
    for (int i = 1; i < 100; ++i)for (int j = 0; j <= i; ++j) {
        if (j == 0)C[i][j] = 1;
        else C[i][j] = C[i - 1][j] + C[i - 1][j - 1];
    }
    int a, b, n;
    cin >> a >> b >> n;
    //for (int mask = 1; mask < (1 << n); ++mask) {
    //    vector<bool> used(2 * n);
    //    for (int i = 0; i < n; ++i) {
    //        if (mask & (1 << i)) {
    //            used[i] = used[i + n] = 1;
    //        }
    //    }
    //    bool ok = 1;
    //    for (int i = find(used.begin(), used.end(), 1) - used.begin();
   i < n; ++i) {
    //        int cnt = 1;
    //        int it = i + 1;
    //        while (!used[it]) {
    //            ++it;
    //            cnt++;
    //        }
    //        i = it - 1;
    //        if (cnt == a || cnt == b)continue;
    //        else ok = 0;
    //    }
    //    if (ok) {
    //        ans++;
    //        //cout << mask << '\n';
    //    }
    //}
    vector<ll> dp(n + 1);
    if (n < a && n < b) {
        cout << "0\n";
        return;
    }
```

```
    dp[0] = 1;
    for (int i = 1; i <= n; ++i) {
        if (i - a >= 0)dp[i] += dp[i - a];
        if (i - b >= 0)dp[i] += dp[i - b];
    }
    ll ans = 0;
    for (int i = 0; i < b; ++i) {
        ans += (i < a ? dp[n - a] : 0) + (b <= n ? dp[n - b] : 0);
    }
    cout << ans << '\n';
}

signed main() {
    ios_base::sync_with_stdio(false);
    cin.tie(0);
    cout.tie(0);
    cout.precision(40);
    int tt = 1;
    //cin >> tt;

    while (tt--) {
        solve();
    }
}
```

Task B

```
/*
-Werror
g++ solution.cpp -o solution.exe -D_GLIBCXX_DEBUG -Wall -Wextra -Wshadow
clang++ solution.cpp -o solution.exe -fsanitize=undefined -fsanitize=
    address -g -g3 -Wall -Wextra
*/

//#ifdef LOCAL  //-DLOCAL
//# debug(x) cerr << __LINE__ << ") " << (#x) << " = " << (x) << endl
//#else
//# debug(x) ;
//#endif

//#pragma GCC optimize("Ofast")//O3
//#pragma GCC target("avx2")

#include <algorithm>
#include <cmath>
#include <fstream>
#include <iostream>
#include <map>
#include <numeric>
#include <queue>
#include <random>
#include <set>
#include <stack>
```

```
#include <string>
#include <unordered_map>
#include <unordered_set>
#include <vector>
#include <iomanip>
#include <bitset>
#include <cassert>
#include <chrono>

using namespace std;
using ld = long double;
using ll = long long;
using ull = unsigned long long;
//using BigInt = __int128_t;
mt19937 rnd(chrono::high_resolution_clock::now().time_since_epoch().count
    ());

ll dist(ll x1, ll y1, ll z1, ll x2, ll y2, ll z2) {
    return (abs(x1 - x2) + abs(y1 - y2) + abs(z1 - z2)) / 2;
}

ll sgn(ll x) {
    if (x < 0)return -1;
    return 1;
}

const int k = 100000000;

void solve() {
    ll n;
    cin >> n;
    ll x1, y1, z1, x2, y2, z2;
    cin >> x1 >> y1 >> z1 >> x2 >> y2 >> z2;
    vector<pair<ll, ll>> t = { {abs(x1), 1}, {abs(y1), 2}, {abs(z1), 3}
   };
    sort(t.rbegin(), t.rend());
    ll c = t[0].second;
    ll ans = dist(x1, y1, z1, x2, y2, z2);
    if (1) {
        ll d = n - abs(x1);
        ll nx = (x1 < 0 ? -n : n);
        ll ny = y1 + sgn(y1) * d / 2;
        ll nz = z1 + sgn(z1) * d / 2;
        ny += sgn(y1) * (d % 2);
        ll r = nx + ny + nz;
        for (int dy = -k; dy <= k; ++dy) {
            ny += dy;
            nz += -dy - r;
            if (-n <= nx && nx <= n && -n <= ny && ny <= n && -n <= nz &&
     nz <= n && nx + ny + nz == 0) {
                ans = min(ans, dist(x1, y1, z1, nx, ny, nz) + dist(x2, y2
    , z2, -nx, -nz, -ny));
            }
```

```
            ny -= dy;
            nz -= (-dy - r);
        }
    }
    if (1) {
        ll d = n - abs(y1);
        ll ny = (y1 < 0 ? -n : n);
        ll nx = x1 + sgn(x1) * d / 2;
        ll nz = z1 + sgn(z1) * d / 2;
        nx += sgn(x1) * (d % 2);
        ll r = nx + ny + nz;
        for (int dx = -k; dx <= k; ++dx) {
            nx += dx;
            nz += -dx - r;
            if (-n <= nx && nx <= n && -n <= ny && ny <= n && -n <= nz &&
     nz <= n && nx + ny + nz == 0) {
                ans = min(ans, dist(x1, y1, z1, nx, ny, nz) + dist(x2, y2
    , z2, -nz, -ny, -nx));
            }
            nx -= dx;
            nz -= (-dx - r);
        }
    }
    if(1){
        ll d = n - abs(z1);
        ll nz = (z1 < 0 ? -n : n);
        ll ny = y1 + sgn(y1) * d / 2;
        ll nx = x1 + sgn(x1) * d / 2;
        nx += sgn(x1) * (d % 2);
        ll r = nx + ny + nz;
        for (int dy = -k; dy <= k; ++dy){
             ny += dy;
             nx += -dy - r;
            if (-n <= nx && nx <= n && -n <= ny && ny <= n && -n <= nz &&
     nz <= n && nx + ny + nz == 0) {
                ans = min(ans, dist(x1, y1, z1, nx, ny, nz) + dist(x2, y2
    , z2, -ny, -nx, -nz));
            }
            ny -= dy;
            nx -= (-dy - r);
        }
    }
    cout << ans << '\n';
}

signed main() {
    ios_base::sync_with_stdio(false);
    cin.tie(0);
    cout.tie(0);
    cout.precision(40);
    int tt = 1;
    //cin >> tt;
```

```
    while (tt--) {
        solve();
    }
}
```

Task C

```
/*
-Werror
g++ solution.cpp -o solution.exe -D_GLIBCXX_DEBUG -Wall -Wextra -Wshadow
clang++ solution.cpp -o solution.exe -fsanitize=undefined -fsanitize=
    address -g -g3 -Wall -Wextra
*/

//#ifdef LOCAL  //-DLOCAL
//# debug(x) cerr << __LINE__ << ") " << (#x) << " = " << (x) << endl
//#else
//# debug(x) ;
//#endif

//#pragma GCC optimize("Ofast")//O3
//#pragma GCC target("avx2")

#include <algorithm>
#include <cmath>
#include <fstream>
#include <iostream>
#include <map>
#include <numeric>
#include <queue>
#include <random>
#include <set>
#include <stack>
#include <string>
#include <unordered_map>
#include <unordered_set>
#include <vector>
#include <iomanip>
#include <bitset>
#include <cassert>
#include <chrono>

using namespace std;
using ld = long double;
using ll = long long;
using ull = unsigned long long;
//using BigInt = __int128_t;
mt19937 rnd(chrono::high_resolution_clock::now().time_since_epoch().count
    ());

const int N = 2e5 + 5;

const ll mod1 = 1e9 + 9, mod2 = 1e9 + 7;
```

```
const ll p = 179;
ll pows1[N];
ll pows2[N];

void solve() {
    pows1[0] = pows2[0] = 1;
    for (int i = 1; i < N; ++i) {
        pows1[i] = (pows1[i - 1] * p) % mod1;
        pows2[i] = (pows2[i - 1] * p) % mod2;
    }
    string tp;
    cin >> tp;
    if (tp[0] == 'f') {
        int n;
        cin >> n;
        ll h1 = 0;
        ll h2 = 0;
        string s;
        cin >> s;
        for (char c : s) {
            h1 = (h1 * p + c) % mod1;
            h2 = (h2 * p + c) % mod2;
        }
        cout << h1 << ' ' << h2 << '\n';
        return;
    }
    int n;
    cin >> n;
    string s;
    cin >> s;
    s += s;
    n += n;
    vector<ll> h1(n + 1), h2(n + 1);
    for (int i = 1; i < n; ++i) {
        h1[i] = ((h1[i - 1] * p) % mod1 + s[i - 1]) % mod1;
        h2[i] = ((h2[i - 1] * p) % mod2 + s[i - 1]) % mod2;
    }
    ll fh1, fh2;
    cin >> fh1 >> fh2;
    n /= 2;
    for (int i = n - 1; i < 2 * n; ++i) {
        ll sh1 = (h1[i + 1] - (h1[i - n + 1] * pows1[n]) % mod1 + mod1) %
     mod1;
        ll sh2 = (h2[i + 1] - (h2[i - n + 1] * pows2[n]) % mod2 + mod2) %
     mod2;
        if (sh1 == fh1 && sh2 == fh2) {
            cout << "YES\n";
            return;
        }
    }
    cout << "NO\n";
}
```

```
signed main() {
    ios_base::sync_with_stdio(false);
    cin.tie(0);
    cout.tie(0);
    cout.precision(40);
    int tt = 1;
    //cin >> tt;

    while (tt--) {
        solve();
    }
}
```

Task D

```
/*
-Werror
g++ solution.cpp -o solution.exe -D_GLIBCXX_DEBUG -Wall -Wextra -Wshadow
clang++ solution.cpp -o solution.exe -fsanitize=undefined -fsanitize=
    address -g -g3 -Wall -Wextra
*/

//#ifdef LOCAL  //-DLOCAL
//# debug(x) cerr << __LINE__ << ") " << (#x) << " = " << (x) << endl
//#else
//# debug(x) ;
//#endif

//#pragma GCC optimize("Ofast")//O3
//#pragma GCC target("avx2")

#include <algorithm>
#include <cmath>
#include <fstream>
#include <iostream>
#include <map>
#include <numeric>
#include <queue>
#include <random>
#include <set>
#include <stack>
#include <string>
#include <unordered_map>
#include <unordered_set>
#include <vector>
#include <iomanip>
#include <bitset>
#include <cassert>
#include <chrono>

using namespace std;
using ld = long double;
using ll = long long;
```

```
using ull = unsigned long long;
//using BigInt = __int128_t;
mt19937 rnd(chrono::high_resolution_clock::now().time_since_epoch().count
    ());

char ask(int x, int y) {

    cout << x + 1 << ' ' << y + 1 << endl;
    char c;
    cin >> c;
    return c;
}

vector<int> perm = { 3, 1 , 2 };

void solve() {
    map<vector<char>, vector<int>> m;
    map<int, vector<pair<int, int>>> Q;
    for (int n = 3; n <= 5; ++n) {
        vector<pair<int, int>> quer(100 * n);
        for (int i = 0; i < 100 * n; ++i)quer[i] = { rnd() % n, rnd() % n
     };
        Q[n] = quer;
        vector<int> p(n);
        iota(p.begin(), p.end(), 0);
        do {
            vector<char> q;
            auto st = p;
            for (int i = 0; i < 100 * n; ++i) {
                auto np = p;
                swap(np[quer[i].first], np[quer[i].second]);
                np.push_back(np[0]);
                np.erase(np.begin());
                if (np == p)q.push_back('=');
                else if (np < p) q.push_back('<');//
                else q.push_back('>');
                p = np;
            }
            m[q] = st;
            p = st;
        } while (next_permutation(p.begin(), p.end()));
    }
    int n;
    cin >> n;
    vector<char> q;
    for (auto i : Q[n]) {
        q.push_back(ask(i.first, i.second));
    }
    auto p = m[q];
    for (int i = 0; i < n; ++i)p[i] = n - p[i] - 1;
    while (1) {
        vector<int> a(n);
        for (int i = 0; i < n; ++i)a[p[i]] = i;
```

```
        bool ok = 1;
        for (int i = 0; i < n; ++i)if (a[i] != i)ok = 0;
        if (ok) {
            while (ask(0, 0) != '!');
            return;
        }
        for (int i = 0; i < n; ++i) {
            if (p[i] != i) {
                int pos = a[i];
                ask(i, pos);
                a[p[i]] = pos;
                a[i] = i;
                swap(p[i], p[pos]);
                break;
            }
        }//
    }
}

signed main() {
    //ios_base::sync_with_stdio(false);
    //cin.tie(0);
    //cout.tie(0);
    //cout.precision(40);
    int tt = 1;
    //cin >> tt;

    while (tt--) {
        solve();
    }
}
```

Task E

```
/*
-Werror
g++ solution.cpp -o solution.exe -D_GLIBCXX_DEBUG -Wall -Wextra -Wshadow
clang++ solution.cpp -o solution.exe -fsanitize=undefined -fsanitize=
    address -g -g3 -Wall -Wextra
*/

//#ifdef LOCAL  //-DLOCAL
//# debug(x) cerr << __LINE__ << ") " << (#x) << " = " << (x) << endl
//#else
//# debug(x) ;
//#endif

//#pragma GCC optimize("Ofast")//O3
//#pragma GCC target("avx2")

#include <algorithm>
#include <cmath>
#include <fstream>
```

```
#include <iostream>
#include <map>
#include <numeric>
#include <queue>
#include <random>
#include <set>
#include <stack>
#include <string>
#include <unordered_map>
#include <unordered_set>
#include <vector>
#include <iomanip>
#include <bitset>
#include <cassert>
#include <chrono>

using namespace std;
using ld = long double;
using ll = long long;
using ull = unsigned long long;
//using BigInt = __int128_t;
mt19937 rnd(chrono::high_resolution_clock::now().time_since_epoch().count
    ());

const int N = 1e5 + 5;
vector<int> gr[N];
int start[N];
int a[N];
int timer = 0;

void dfs(int v) {
    if (timer >= N)return;
    a[timer] = v;
    int n = gr[v].size();
    int u = gr[v][start[v]];
    start[v]++;
    start[v] %= n;
    ++timer;
    dfs(u);
}

void solve() {
    int n, q;
    cin >> n >> q;
    for (int i = 0; i < n; ++i) {
        int d;
        cin >> d;
        gr[i].resize(d);
        for (int j = 0; j < d; ++j) {
            int u;
            cin >> u;
            gr[i][j] = u - 1;
        }
```

```
    }
    dfs(0);
    for (int i = 0; i < q; ++i) {
        int x;
        cin >> x;
        cout << a[x] + 1 << '\n';
    }
}

signed main() {
    ios_base::sync_with_stdio(false);
    cin.tie(0);
    cout.tie(0);
    cout.precision(40);
    int tt = 1;
    //cin >> tt;

    while (tt--) {
        solve();
    }
}
```

Task F

```
/*
-Werror
g++ solution.cpp -o solution.exe -D_GLIBCXX_DEBUG -Wall -Wextra -Wshadow
clang++ solution.cpp -o solution.exe -fsanitize=undefined -fsanitize=
    address -g -g3 -Wall -Wextra
*/

//#ifdef LOCAL  //-DLOCAL
//# debug(x) cerr << __LINE__ << ") " << (#x) << " = " << (x) << endl
//#else
//# debug(x) ;
//#endif

//#pragma GCC optimize("Ofast")//O3
//#pragma GCC target("avx2")

#include <algorithm>
#include <cmath>
#include <fstream>
#include <iostream>
#include <map>
#include <numeric>
#include <queue>
#include <random>
#include <set>
#include <stack>
#include <string>
#include <unordered_map>
#include <unordered_set>
```

```
#include <vector>
#include <iomanip>
#include <bitset>
#include <cassert>
#include <chrono>

using namespace std;
using ld = long double;
using ll = long long;
using ull = unsigned long long;
//using BigInt = __int128_t;
mt19937 rnd(chrono::high_resolution_clock::now().time_since_epoch().count
    ());

void solve() {
    int h, w;
    cin >> h >> w;
    vector<string> a(h);
    for (auto& i : a)cin >> i;
    vector<vector<char>> res1(h, vector<char>(w));
    vector<vector<char>> res2(h, vector<char>(w));
    ll ans1 = 0;
    ll ans2 = 0;
    for (int i = 0; i < h; ++i) {
        for (int j = 0; j < w - 1; j += 2) {
            //cout << "L.";
            res1[i][j] = 'L';
            res1[i][j + 1] = 'R';
            ans1 += (a[i][j] - '0') * (a[i][j + 1] - '0');
        }
        if (w % 2) {
            if (i % 2 == 0) {
                res1[i][w - 1] = 'T';
                ans1 += (a[i][w - 1] - '0') * (a[i + 1][w - 1] - '0');
            }
            else{
                res1[i][w - 1] = 'B';
            }
        }
    }
    for (int i = 0; i < h; i += 2) {
        for (int j = 0; j < w; ++j) {
            if (i + 1 < h) {
                res2[i][j] = 'T';
                res2[i + 1][j] = 'B';
                ans2 += (a[i][j] - '0') * (a[i + 1][j] - '0');
            }
            else {
                if ((j == 0 || res2[i][j] == '.') && j + 1 < w) {
                    res2[i][j] = 'L';
                    ans2 += (a[i][j] - '0') * (a[i][j + 1] - '0');
                }
                else {
```

```
                    if (j != 0 && res2[i][j - 1] == 'L')res2[i][j] = 'R';
                    else res2[i][j] = 'L';
                }
            }
        }
    }
    if (ans1 >= ans2) {
        cout << ans1 << '\n';
        for (int i = 0; i < h; ++i) {
            for (char c : res1[i])cout << c;
            cout << '\n';
        }
    }
    else {
        cout << ans2 << '\n';
        for (int i = 0; i < h; ++i) {
            for (char c : res2[i])cout << c;
            cout << '\n';
        }
    }
}

signed main() {
    //ios_base::sync_with_stdio(false);
    //cin.tie(0);
    //cout.tie(0);
    //cout.precision(40);
    int tt = 1;
    //cin >> tt;

    while (tt--) {
        solve();
    }
}
```