Олимпиада СПбГУ
по информатике
2023/24 учебного года

24fin0462 – Тырин Михаил Павлович

A	B	C	D	E	F	sum
100	100	100	100	20	91	511

Task A

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

using namespace std;

mt19937 rnd(chrono::high_resolution_clock::now().time_since_epoch().count
    ());

#ifndef LOCAL
// #pragma GCC optimize("unroll-loops,inline,Ofast")
// #pragma GCC target("avx2")
#endif

#define int long long
#define all(v) (v).begin(), (v).end()
#define rall(v) (v).rbegin(), (v).rend()
#define pint pair<int, int>
#define endl '\n'

template<class T>
bool cmax(T& a, T b) {
    if (a >= b) return false;
    a = b;
    return true;
}

template<class T>
bool cmin(T& a, T b) {
    if (a <= b) return false;
    a = b;
```

```
    return true;
}

inline void solve() {
    int a, b, n;
    cin >> a >> b >> n;
    vector<int> dp(n + 1, 0);
    dp[0] = 1;
    for (int i = 1; i <= n; ++i) {
        if (i - a >= 0) dp[i] += dp[i - a];
        if (i - b >= 0 && a != b) dp[i] += dp[i - b];
    }
    int rez = 0;
    if (a <= n) rez += dp[n - a] * a;
    if (b <= n && a != b) rez += dp[n - b] * b;
    cout << rez << endl;
}

signed main() {
    cin.tie(nullptr);
    cout.tie(nullptr);
    ios_base::sync_with_stdio(false);
#ifdef LOCAL
    freopen("input.txt", "r", stdin);
    freopen("output.txt", "w", stdout);
#endif
    int ttt = 1;
    // cin >> ttt;
    while (ttt--) {
        solve();
    }
}
```

Task B

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

using namespace std;

mt19937 rnd(chrono::high_resolution_clock::now().time_since_epoch().count
    ());

#ifndef LOCAL
// #pragma GCC optimize("unroll-loops,inline,Ofast")
// #pragma GCC target("avx2")
#endif

#define int long long
#define all(v) (v).begin(), (v).end()
#define rall(v) (v).rbegin(), (v).rend()
#define pint pair<int, int>
#define endl '\n'
```

```
template<class T>
bool cmax(T& a, T b) {
    if (a >= b) return false;
    a = b;
    return true;
}

template<class T>
bool cmin(T& a, T b) {
    if (a <= b) return false;
    a = b;
    return true;
}

inline void solve() {
    int n;
    cin >> n;
    int x1, x2, y1, y2, z1, z2;
    cin >> x1 >> y1 >> z1 >> x2 >> y2 >> z2;
    auto dist = [](int x1, int y1, int z1, int x2, int y2, int z2) {
        return (abs(x1 - x2) + abs(y1 - y2) + abs(z1 - z2)) / 2;
    };
    int mn = dist(x1, y1, z1, x2, y2, z2);
    auto go = [&](int x0, int y0, int z0) {
        x0 += x2;
        y0 += y2;
        z0 += z2;
        cmin(mn, dist(x1, y1, z1, x0, y0, z0));
//        cout << mn << endl;
    };
//    cout << mn << endl;
    go(-n, -n, 2 * n);
    go(n, n, -2 * n);
    go(2 * n, -n, -n);
    go(-2 * n, n, n);
    go(-n, 2 * n, -n);
    go(n, -2 * n, n);
    cout << mn << endl;
}

signed main() {
    cin.tie(nullptr);
    cout.tie(nullptr);
    ios_base::sync_with_stdio(false);
#ifdef LOCAL
    freopen("input.txt", "r", stdin);
    freopen("output.txt", "w", stdout);
#endif
    int ttt = 1;
    // cin >> ttt;
    while (ttt--) {
        solve();
    }
```

```
}
```

Task C

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

using namespace std;

mt19937 rnd(chrono::high_resolution_clock::now().time_since_epoch().count
    ());

#ifndef LOCAL
// #pragma GCC optimize("unroll-loops,inline,Ofast")
// #pragma GCC target("avx2")
#endif

#define int long long
#define all(v) (v).begin(), (v).end()
#define rall(v) (v).rbegin(), (v).rend()
#define pint pair<int, int>
#define endl '\n'

template<class T>
bool cmax(T& a, T b) {
    if (a >= b) return false;
    a = b;
    return true;
}

template<class T>
bool cmin(T& a, T b) {
    if (a <= b) return false;
    a = b;
    return true;
}

inline void solve() {
    string s;
    cin >> s;
    int n;
    cin >> n;
    string a;
    cin >> a;
    int mod = 998244353;
    int p = 97;
    if (s == "first") {
        vector<int> hashes;
        for (int i = 0; i < 10; ++i) {
            int len = n / 10;
            int hash = 0;
            for (int j = i * len; j - i * len < len; ++j) {
                hash = hash * p + a[j];
                hash %= mod;
```

```
            }
            hashes.push_back(hash);
        }
        {
            int len = n % 10;
            int hash = 0;
            for (int j = n - len; j < n; ++j) {
                hash = hash * p + a[j];
                hash %= mod;
            }
            hashes.push_back(hash);
        }
        for (auto& el : hashes) {
            for (int j = 8; j >= 0; --j) {
                cout << el / (int)pow(10, j) % 10;
            }
        }
    } else {
        vector<int> hashes;
        cin >> s;
//        cout << s.size() << endl;
        for (int j = 0; j < 99; ++j) {
            if (j % 9 == 0) hashes.emplace_back();
            hashes.back() = hashes.back() * 10 + s[j] - '0';
        }
        a += a;
        vector<int> hash(2 * n + 1);
        vector<int> ppow(2 * n + 1, 1);
        for (int i = 0; i < 2 * n; ++i) {
            hash[i + 1] = hash[i] * p + a[i];
            hash[i + 1] %= mod;
            ppow[i + 1] = ppow[i] * p;
            ppow[i + 1] %= mod;
        }
        for (int start = 0; start < n; ++start) {
            vector<int> have;
            for (int i = 0; i < 10; ++i) {
                int len = n / 10;
                int hsh = ((hash[i * len + start + len] - hash[i * len +
    start] * ppow[len] % mod) % mod + mod) % mod;
                have.push_back(hsh);
            }
            {
                int len = n % 10;
                int hsh = ((hash[n - len + start + len] - hash[n - len +
    start] * ppow[len] % mod) % mod + mod) % mod;
                have.push_back(hsh);
            }
//            for (auto& el : have) cout << el << ' ';
//            cout << endl;
            if (have == hashes) {
                cout << "yes\n";
                return;
```

```
            }
        }
        cout << "no\n";
    }
}

signed main() {
    cin.tie(nullptr);
    cout.tie(nullptr);
    ios_base::sync_with_stdio(false);
#ifdef LOCAL
    freopen("input.txt", "r", stdin);
    freopen("output.txt", "w", stdout);
#endif
    int ttt = 1;
    // cin >> ttt;
    while (ttt--) {
        solve();
    }
}
```

Task D

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

using namespace std;

mt19937 rnd(chrono::high_resolution_clock::now().time_since_epoch().count
    ());

#ifndef LOCAL
// #pragma GCC optimize("unroll-loops,inline,Ofast")
// #pragma GCC target("avx2")
#endif

#define int long long
#define all(v) (v).begin(), (v).end()
#define rall(v) (v).rbegin(), (v).rend()
#define pint pair<int, int>
#define endl '\n'

template<class T>
bool cmax(T& a, T b) {
    if (a >= b) return false;
    a = b;
    return true;
}

template<class T>
bool cmin(T& a, T b) {
    if (a <= b) return false;
    a = b;
    return true;
```

```
}

inline void solve() {
    int n;
    cin >> n;
    vector<int> inds(n);
    set<int> more;
    vector<int> now(n);
    iota(all(now), 0);
    int have = 0;
    while (have < n) {
        set<int> checked;
        // first - max
        int need = n - have - 1;
        while (more.count(now[0])) {
            cout << "1 1\n";
            cout.flush();
            char c;
            cin >> c;
            if (c == '!') return;
            now.push_back(now[0]);
            now.erase(now.begin());
        }
        while (checked.size() < need) {
            if (more.count(now[1]) || checked.count(now[1])) {
                cout << "1 2\n";
                cout.flush();
                swap(now[0], now[1]);
                char c;
                cin >> c;
                if (c == '!') return;
                now.push_back(now[0]);
                now.erase(now.begin());
                continue;
            }
            cout << "1 1\n";
            cout.flush();
            now.push_back(now[0]);
            now.erase(now.begin());
            char c;
            cin >> c;
            if (c == '!') return;
            if (c == '>') {
                checked.insert(now.back());
                continue;
            }
            checked.insert(now[0]);
            cout << "2 " + to_string(n) << endl;
            cout.flush();
            swap(now[1], now.back());
            cin >> c;
            if (c == '!') return;
            now.push_back(now[0]);
```

```
        now.erase(now.begin());
    }
    inds[n - have - 1] = now[0];
    more.insert(now[0]);
    have++;
    cout << "1 1\n";
    cout.flush();
    char c;
    cin >> c;
    if (c == '!') return;
    now.push_back(now[0]);
    now.erase(now.begin());
}
vector<int> rinds(n);
for (int i = 0; i < n; ++i) {
    rinds[inds[i]] = i;
}
for (int i = 0; i < n; ++i) {
    now[i] = rinds[now[i]];
}
for (int i = 0; i < n; ++i) {
    inds[now[i]] = i;
}
int to = n - 2;
while (to >= 0 && (inds[to + 1] + 1) % n == inds[to]) to--;
if (to == -1) {
    while (true) {
        cout << "1 1\n";
        cout.flush();
        char c;
        cin >> c;
        if (c == '!') return;
    }
}
while (to >= 0) {
    if ((inds[to + 1] + 1) % n == inds[to]) {
        to--;
        continue;
    }
    cout << (inds[to + 1] + 1) % n + 1 << ' ' << inds[to] + 1 <<endl;
    cout.flush();
    int t1 = (inds[to + 1] + 1) % n;
    int t2 = inds[to];
    swap(inds[now[(inds[to + 1] + 1) % n]], inds[to]);
    swap(now[t1], now[t2]);
    char c;
    cin >> c;
    if (c == '!') return;
    now.push_back(now[0]);
    now.erase(now.begin());
    for (auto& el : inds) {
        el = (el - 1 + n) % n;
    }
```

```
        to--;
    }
    while (true) {
        cout << "1 1\n";
        cout.flush();
        char c;
        cin >> c;
        if (c == '!') return;
    }
}

signed main() {
//    cin.tie(nullptr);
//    cout.tie(nullptr);
//    ios_base::sync_with_stdio(false);
#ifdef LOCAL
//    freopen("input.txt", "r", stdin);
//    freopen("output.txt", "w", stdout);
#endif
    int ttt = 1;
    // cin >> ttt;
    while (ttt--) {
        solve();
    }
}
```

Task E

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

using namespace std;

mt19937 rnd(chrono::high_resolution_clock::now().time_since_epoch().count
    ());

#ifndef LOCAL
// #pragma GCC optimize("unroll-loops,inline,Ofast")
// #pragma GCC target("avx2")
#endif

#define int long long
#define all(v) (v).begin(), (v).end()
#define rall(v) (v).rbegin(), (v).rend()
#define pint pair<int, int>
#define endl '\n'

template<class T>
bool cmax(T& a, T b) {
    if (a >= b) return false;
    a = b;
    return true;
}
```

```
template<class T>
bool cmin(T& a, T b) {
    if (a <= b) return false;
    a = b;
    return true;
}

inline void solve() {
    int n;
    cin >> n;
    vector<vector<int>> graph(n);
    int q;
    cin >> q;
    for (int i = 0; i < n; ++i) {
        int c;
        cin >> c;
        while (c--) {
            int x;
            cin >> x;
            x--;
            graph[i].push_back(x);
        }
    }
    vector<pint> que(q);
    for (int i = 0; i < q; ++i) {
        cin >> que[i].first;
        que[i].second = i;
    }
    sort(all(que));
    int ind = 0;
    int v = 0;
    vector<int> c_i(n, 0);
    vector<int> ans(q);
    int time = 0;
    while (ind != q) {
        while (ind < q && que[ind].first == time) {
            ans[que[ind++].second] = v;
        }
        time++;
        int u = v;
        v = graph[v][c_i[v]];
        c_i[u] = (c_i[u] + 1) % graph[u].size();
    }
    for (auto& el : ans) cout << el + 1 << ' ';
}

signed main() {
    cin.tie(nullptr);
    cout.tie(nullptr);
    ios_base::sync_with_stdio(false);
#ifdef LOCAL
    freopen("input.txt", "r", stdin);
    freopen("output.txt", "w", stdout);
```

```
#endif
    int ttt = 1;
    // cin >> ttt;
    while (ttt--) {
        solve();
    }
}
```

Task F

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

using namespace std;

mt19937 rnd(chrono::high_resolution_clock::now().time_since_epoch().count
    ());

#ifndef LOCAL
#pragma GCC optimize("unroll-loops,inline,Ofast")
// #pragma GCC target("avx2")
#endif

// #define int long long
#define all(v) (v).begin(), (v).end()
#define rall(v) (v).rbegin(), (v).rend()
#define pint pair<int, int>
#define endl '\n'

template<class T>
bool cmax(T& a, T b) {
    if (a >= b) return false;
    a = b;
    return true;
}

template<class T>
bool cmin(T& a, T b) {
    if (a <= b) return false;
    a = b;
    return true;
}

inline pair<int, vector<vector<char>>> rsolve(int h, int w, vector<string
    > v) {
    vector<vector<pint>> z(10);
    for (int i = 0; i < h; ++i) {
        for (int j = 0; j < w; ++j) {
            z[v[i][j] - '0'].emplace_back(i, j);
        }
    }
    vector<vector<char>> used(h, vector<char>(w, '.'));
    int r = 0;
```

```
    for (int k = 9; k >= 0; --k) {
        for (auto& [a, b] : z[k]) {
            if (used[a][b] != '.') continue;
            int mx = -1, ma = a, mb = b;
            int tp = -1;
            {
                int c = (a - 1 + h) % h, d = b;
                if (used[c][d] == '.') if (cmax<int>(mx, v[c][d] - '0'))
    ma = c, mb = d, tp = 1;
            }
            {
                int c = (a + 1) % h, d = b;
                if (used[c][d] == '.') if (cmax<int>(mx, v[c][d] - '0'))
    ma = c, mb = d, tp = 2;
            }
            {
                int c = a, d = (b - 1 + w) % w;
                if (used[c][d] == '.') if (cmax<int>(mx, v[c][d] - '0'))
    ma = c, mb = d, tp = 3;
            }
            {
                int c = a, d = (b + 1) % w;
                if (used[c][d] == '.') if (cmax<int>(mx, v[c][d] - '0'))
    ma = c, mb = d, tp = 4;
            }
            if (mx != -1) {
                r += k * mx;
                switch(tp) {
                    case 1:
                        used[a][b] = 'B', used[ma][mb] = 'T';
                        break;
                    case 2:
                        used[a][b] = 'T', used[ma][mb] = 'B';
                        break;
                    case 3:
                        used[a][b] = 'R', used[ma][mb] = 'L';
                        break;
                    case 4:
                        used[a][b] = 'L', used[ma][mb] = 'R';
                        break;
                }
            }
        }
    }
    return {r, used};
}

inline pair<int, vector<vector<char>>> solve(int h, int w, vector<string>
     v) {
    vector<vector<pint>> z(10);
    for (int i = 0; i < h; ++i) {
        for (int j = 0; j < w; ++j) {
            z[v[i][j] - '0'].emplace_back(i, j);
```

```
    }
}
vector<vector<char>> used(h, vector<char>(w, '.'));
int r = 0;
for (int k = 9; k >= 0; --k) {
    shuffle(all(z[k]), rnd);
    for (auto& [a, b] : z[k]) {
        if (used[a][b] != '.') continue;
        vector<pint> best;
        {
            int c = (a - 1 + h) % h, d = b;
            if (used[c][d] == '.') best.emplace_back(v[c][d] - '0',
1);
        }
        {
            int c = (a + 1) % h, d = b;
            if (used[c][d] == '.') best.emplace_back(v[c][d] - '0',
2);
        }
        {
            int c = a, d = (b - 1 + w) % w;
            if (used[c][d] == '.') best.emplace_back(v[c][d] - '0',
3);
        }
        {
            int c = a, d = (b + 1) % w;
            if (used[c][d] == '.') best.emplace_back(v[c][d] - '0',
4);
        }
        if (best.empty()) continue;
        sort(rall(best));
        int got = rnd() % min<int>(2, best.size());
        int tp = best[got].second;
        r += best[got].first * k;
        int ma = a, mb = b;
        switch(tp) {
            case 1:
                ma = (a - 1 + h) % h;
                used[a][b] = 'B', used[ma][mb] = 'T';
                break;
            case 2:
                ma = (a + 1) % h;
                used[a][b] = 'T', used[ma][mb] = 'B';
                break;
            case 3:
                mb = (b - 1 + w) % w;
                used[a][b] = 'R', used[ma][mb] = 'L';
                break;
            case 4:
                mb = (b + 1) % w;
                used[a][b] = 'L', used[ma][mb] = 'R';
                break;
        }
```

```
        }
    }
    return {r, used};
}

signed main() {
    cin.tie(nullptr);
    cout.tie(nullptr);
    ios_base::sync_with_stdio(false);
#ifdef LOCAL
    freopen("input.txt", "r", stdin);
    freopen("output.txt", "w", stdout);
#endif
    int ttt = 1;
    // cin >> ttt;
    while (ttt--) {
        int w, h;
        cin >> h >> w;
        vector<string> v(h);
        for (auto& el : v) cin >> el;
        int best;
        vector<vector<char>> rez;
        tie(best, rez) = rsolve(h, w, v);
        while ((double)clock() / CLOCKS_PER_SEC <= 1.95) {
            int now;
            vector<vector<char>> new_;
            tie(now, new_) =  solve(h, w, v);
            if (cmax(best, now)) {
                rez = std::move(new_);
            }
        }
        cout << best << endl;
        for (auto& el : rez) {
            for (auto& x : el) cout << x;
            cout << endl;
        }
    }
}
```