Олимпиада СПбГУ
по информатике
2023/24 учебного года

24fin0118 – Грек Фёдор Александрович

A	B	C	D	E	F	sum
100	100	100	35	20	61	416

Task A

```
#include <bits/stdc++.h>
#include <ext/pb_ds/assoc_container.hpp>
#include <ext/pb_ds/tree_policy.hpp>

using namespace std;
using namespace __gnu_pbds;

//#pragma GCC optimize("O3,unroll-loops")
//#pragma GCC target("avx2,bmi,bmi2,lzcnt,popcnt")

#define ll long long
#define ld long double
#define klm ios_base::sync_with_stdio(0); cin.tie(0); cout.tie(0)
#define all(a) (a).begin(), (a).end()
#define pb(a) push_back(a)
#define sz(a) (long long)a.size()

mt19937_64 rnd;

void solve() {
    ll a, b, n, ans = 0;
    cin >> a >> b >> n;
    vector<ll>dp(n + 2, 0), banned(n + 1, 1);
    for(int j = 0; j < n; ++j){
        dp[0] = 1;
        for(int i = 0; i < n; ++i){
            if (i + a <= n){
                dp[i + a] += banned[i + a] * dp[i];
            }
```

```
            if (i + b <= n){
                dp[i + b] += banned[i + b] * dp[i];
            }
        }
        ans += dp[n];
        banned[j + 1] = 0;
        fill(all(dp), 0);
    }
    cout << ans << '\n';
}

signed main() {
    int t = 1;
    klm;

    #ifdef LOCAL

    freopen("D:/broshka/input.txt", "r", stdin);
    freopen("D:/broshka/output.txt", "w", stdout);

    #endif
    //cin >> t;
    while (t--) {
        solve();
    }
    return 0;
}
```

Task B

```
#include <bits/stdc++.h>
#include <ext/pb_ds/assoc_container.hpp>
#include <ext/pb_ds/tree_policy.hpp>

using namespace std;
using namespace __gnu_pbds;

//#pragma GCC optimize("O3,unroll-loops")
//#pragma GCC target("avx2,bmi,bmi2,lzcnt,popcnt")

#define ll long long
#define ld long double
#define klm ios_base::sync_with_stdio(0); cin.tie(0); cout.tie(0)
#define all(a) (a).begin(), (a).end()
#define pb(a) push_back(a)
#define sz(a) (long long)a.size()

mt19937_64 rnd;

struct point{
    ll x, y, z;

    point(){
```

```
    }

    point(ll _x, ll _y, ll _z){
        x = _x;
        y = _y;
        z = _z;
    }
};

point operator-(point a, point b){
    return point(a.x - b.x, a.y - b.y, a.z - b.z);
}

point operator+(point a, point b){
    return point(a.x + b.x, a.y + b.y, a.z + b.z);
}

ll dist(point a){
    return (abs(a.x) + abs(a.y) + abs(a.z)) / 2;
}

void solve() {
    ll n;
    cin >> n;

    point a, b, c = {2 * n, -n, -n};
    cin >> a.x >> a.y >> a.z;
    cin >> b.x >> b.y >> b.z;

    ll ans = dist(a - b);
    point a1 = a + c, b1 = b + c;
    ans = min({ans, dist(a1 - b), dist(b1 - a)});

    c = {-2 * n, n, n};
    a1 = a + c;
    b1 = b + c;
    ans = min({ans, dist(a1 - b), dist(b1 - a)});

    c = {n, -2 * n, n};
    a1 = a + c;
    b1 = b + c;
    ans = min({ans, dist(a1 - b), dist(b1 - a)});

    c = {-n, 2 * n, -n};
    a1 = a + c;
    b1 = b + c;
    ans = min({ans, dist(a1 - b), dist(b1 - a)});

    c = {n, n, -2 * n};
    a1 = a + c;
    b1 = b + c;
    ans = min({ans, dist(a1 - b), dist(b1 - a)});
```

```
    c = {-n, -n, 2 * n};
    a1 = a + c;
    b1 = b + c;
    ans = min({ans, dist(a1 - b), dist(b1 - a)});

    cout << ans;
}

signed main() {
    int t = 1;
    klm;

    #ifdef LOCAL

    freopen("D:/broshka/input.txt", "r", stdin);
    freopen("D:/broshka/output.txt", "w", stdout);

    #endif
    //cin >> t;
    while (t--) {
        solve();
    }
    return 0;
}
```

Task C

```
#include <bits/stdc++.h>
#include <ext/pb_ds/assoc_container.hpp>
#include <ext/pb_ds/tree_policy.hpp>

using namespace std;
using namespace __gnu_pbds;

//#pragma GCC optimize("O3,unroll-loops")
//#pragma GCC target("avx2,bmi,bmi2,lzcnt,popcnt")

#define ll long long
#define ld long double
#define klm ios_base::sync_with_stdio(0); cin.tie(0); cout.tie(0)
#define all(a) (a).begin(), (a).end()
#define pb(a) push_back(a)
#define sz(a) (long long)a.size()

mt19937_64 rnd;

void solve() {
    string s, bro;
    cin >> s;
    ll n, p = 73, mod = 1e9 + 7;
    cin >> n;
    cin >> bro;
```

```
    if (s == "first"){
        s = bro;
        ll cur = (ll)s[0];
        for(int i = 1; i < n; ++i){
            cur = (cur * p + (ll)s[i]) % mod;
        }
        cout << cur;
    } else{
        s = bro;
        ll cnt;
        cin >> cnt;
        vector<ll>power(n + 1, 1), hsh(n, 0);
        hsh[0] = (ll)s[0];
        for(int i = 1; i < n; ++i){
            power[i] = (power[i - 1] * p) % mod;
            hsh[i] = (hsh[i - 1] * p + (ll)s[i]) % mod;
        }
        bool ok = (hsh[n - 1] == cnt);
        for(int i = 1; i < n; ++i){
            ll cur = hsh[i - 1], cur2 = (hsh[n - 1] - (hsh[i - 1] * power
    [n - i]) % mod + mod) % mod;
            ll cur1 = ((cur2 * power[i]) % mod + cur) % mod;
            ok |= (cur1 == cnt);
        }
        (ok) ? cout << "YES\n" : cout << "NO\n";
    }
}

signed main() {
    int t = 1;
    klm;

    #ifdef LOCAL

    freopen("D:/broshka/input.txt", "r", stdin);
    freopen("D:/broshka/output.txt", "w", stdout);

    #endif
    //cin >> t;
    while (t--) {
        solve();
    }
    return 0;
}
```

Task D

```
#include <bits/stdc++.h>
#include <ext/pb_ds/assoc_container.hpp>
#include <ext/pb_ds/tree_policy.hpp>

using namespace std;
using namespace __gnu_pbds;
```

```
//#pragma GCC optimize("O3,unroll-loops")
//#pragma GCC target("avx2,bmi,bmi2,lzcnt,popcnt")

#define ll long long
#define ld long double
#define klm ios_base::sync_with_stdio(0); cin.tie(0); cout.tie(0)
#define all(a) (a).begin(), (a).end()
#define pb(a) push_back(a)
#define sz(a) (long long)a.size()

mt19937_64 rnd;

void solve() {
    ll n;
    cin >> n;
    while (true){
        ll i = rnd() % n + 1, j = rnd() % n + 1;
        cout << i << " " << j << endl;
        char x;
        cin >> x;
        if (x == '!'){
            return;
        }
    }
}

signed main() {
    int t = 1;
    klm;

    #ifdef LOCAL

    freopen("D:/broshka/input.txt", "r", stdin);
    freopen("D:/broshka/output.txt", "w", stdout);

    #endif
    //cin >> t;
    while (t--) {
        solve();
    }
    return 0;
}
```

Task E

```
#include <bits/stdc++.h>
#include <ext/pb_ds/assoc_container.hpp>
#include <ext/pb_ds/tree_policy.hpp>

using namespace std;
using namespace __gnu_pbds;
```

```
//#pragma GCC optimize("O3,unroll-loops")
//#pragma GCC target("avx2,bmi,bmi2,lzcnt,popcnt")

#define ll long long
#define ld long double
#define klm ios_base::sync_with_stdio(0); cin.tie(0); cout.tie(0)
#define all(a) (a).begin(), (a).end()
#define pb(a) push_back(a)
#define sz(a) (long long)a.size()

mt19937_64 rnd;

struct negry{
    vector<ll>v;
    ll cur = 0;

    negry(){}

    negry(vector<ll> &v1){
        v = v1;
    }

    ll go(){
        ll k = v[cur];
        cur = (cur + 1) % sz(v);
        return k;
    }
};

vector<negry>g;

void solve() {
    ll n, q;
    cin >> n >> q;
    g.assign(n, negry());

    for(int i = 0; i < n; ++i){
        ll k;
        cin >> k;
        vector<ll>bro(k);
        for(int j = 0; j < k; ++j){
            cin >> bro[j];
            --bro[j];
        }
        g[i] = negry(bro);
    }
    ll cur = 0, pr = 0;
    while(q--){
        ll x;
        cin >> x;
        while(pr < x){
            cur = g[cur].go();
            ++pr;
```

```
        }
        cout << cur + 1 << " ";
    }
}

signed main() {
    int t = 1;
    klm;

    #ifdef LOCAL

    freopen("D:/broshka/input.txt", "r", stdin);
    freopen("D:/broshka/output.txt", "w", stdout);

    #endif
    //cin >> t;
    while (t--) {
        solve();
    }
    return 0;
}
```

Task F

```
#include <bits/stdc++.h>
#include <ext/pb_ds/assoc_container.hpp>
#include <ext/pb_ds/tree_policy.hpp>

using namespace std;
using namespace __gnu_pbds;

//#pragma GCC optimize("O3,unroll-loops")
//#pragma GCC target("avx2,bmi,bmi2,lzcnt,popcnt")

#define ll long long
#define ld long double
#define klm ios_base::sync_with_stdio(0); cin.tie(0); cout.tie(0)
#define all(a) (a).begin(), (a).end()
#define pb(a) push_back(a)
#define sz(a) (long long)a.size()

mt19937_64 rnd;

void solve() {
    ll n, m;
    cin >> n >> m;
    vector<vector<ll>>v(n, vector<ll>(m, 0));
    for(int i = 0; i < n; ++i){
        for(int j = 0; j < m; ++j){
            char x;
            cin >> x;
            v[i][j] = (x - '0');
        }
```

```
}

ll ans = 0;
vector<vector<char>>gg(n, vector<char>(m, '.')), prgg;
prgg = gg;
ld t = 1000000, prt = 1000000;
for(int kkk = 0; kkk < min(3000011, 75000000 / (n * m)); ++kkk){
    auto cur1 = prgg;
    ll cur = 0;
    for(int i = 0; i < n; ++i){
        for(int j = 0; j < m; ++j){
            if (v[i][j] == 0){
                continue;
            }
            if (cur1[i][j] != '.'){
                continue;
            }
            ll bro = rnd() % 100;
            if (bro < 95){
                ll i1 = (i + n - 1) % n, j1 = (j + m - 1) % m;
                if (cur1[i1][j] == '.' && v[i1][j] != 0){
                    if (cur1[i][j1] != '.' || v[i][j1] == 0){
                        cur1[i1][j] = 'T';
                        cur1[i][j] = 'B';
                        cur += v[i][j] * v[i1][j];
                    } else if (v[i1][j] > v[i][j1]){
                        cur1[i1][j] = 'T';
                        cur1[i][j] = 'B';
                        cur += v[i][j] * v[i1][j];
                    } else{
                        cur1[i][j1] = 'L';
                        cur1[i][j] = 'R';
                        cur += v[i][j] * v[i][j1];
                    }
                } else if (cur1[i][j1] == '.' && v[i][j1] != 0){
                    cur1[i][j1] = 'L';
                    cur1[i][j] = 'R';
                    cur += v[i][j] * v[i][j1];
                }
            } else{
                bro = rnd() % 2;
                if (bro){
                    ll i1 = (i + n - 1) % n;
                    if (cur1[i1][j] == '.' && v[i1][j] != 0){
                        cur1[i1][j] = 'T';
                        cur1[i][j] = 'B';
                        cur += (v[i1][j] * v[i][j]);
                    }
                } else{
                    ll j1 = (j + m - 1) % m;
                    if (cur1[i][j1] == '.' && v[i][j1] != 0){
                        cur1[i][j1] = 'L';
                        cur1[i][j] = 'R';
```

```
                                cur += (v[i][j1] * v[i][j]);
                            }
                        }
                    }
                }
            }
            if (cur > ans || rnd() < exp((ans - cur) * prt / t)){
                gg = cur1;
                ans = cur;
            }
            t *= 0.999999;
        }
        cout << ans << '\n';
        for(int i = 0; i < n; ++i){
            for(int j = 0; j < m; ++j){
                cout << gg[i][j];
            }
            cout << '\n';
        }
}

signed main() {
    int t = 1;
    klm;

    #ifdef LOCAL

    freopen("D:/broshka/input.txt", "r", stdin);
    freopen("D:/broshka/output.txt", "w", stdout);

    #endif
    //cin >> t;
    while (t--) {
        solve();
    }
    return 0;
}
```