Олимпиада СПбГУ
по информатике
2023/24 учебного года

24fin0405 – Сапегин Андрей Андреевич

A	B	C	D	E	F	sum
100	100	100	100	20	75	495

Task A

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

using namespace std;

typedef long long ll;
typedef long double ld;

ll cnt = 0;
void rec(int n, int a, int b)
{
    if (n == 0) cnt++;
    if (n <= 0) return;

    rec(n - a, a, b);
    rec(n - b, a, b);
}

void solve()
{
    int a, b, n;
    cin >> a >> b >> n;
    ll ans = 0;
    for (ll cnt1 = 0; cnt1 * a <= n; cnt1++)
    {
        ll cnt2 = n - cnt1 * a;
        if (cnt2 % b) continue;
        cnt2 /= b;
        ll f = 1;
        ll del = 2;
```

```
        for (ll i = cnt2 + 1; i <= cnt2 + cnt1 - 1; i++)
        {
            f *= i;
            while (del <= cnt1 - 1 && f % del == 0)
            {
                f /= del;
                del++;
            }
        }
        ll f2 = 1;
        del = 2;
        for (ll i = cnt1 + 1; i <= cnt1 + cnt2 - 1; i++)
        {
            f2 *= i;
            while (del <= cnt2 - 1 && f2 % del == 0)
            {
                f2 /= del;
                del++;
            }
        }
        // cout << cnt1 << ' ' << cnt2 << endl;
        if (cnt1 > 0)
        {
            ans += f * a;
            // cout << f << ' ' << a << ' ' << f * a << endl;
        }
        if (cnt2 > 0)
        {
            ans += f2 * b;
            // cout << f2 << ' ' << b << ' ' << f2 * b << endl;
        }
        // cout << endl;
    }
    cout << ans;
}

signed main()
{
    ios_base::sync_with_stdio(0);
    cin.tie(0);
    cin.tie(0);

    int t = 1;
    // cin >> t;

    while (t--)
    {
        solve();
    }
    return 0;
}
```

Task B

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

using namespace std;

typedef long long ll;
typedef long double ld;

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

void solve()
{
    ll n;
    cin >> n;
    ll x1, y1, z1;
    cin >> x1 >> y1 >> z1;
    ll x2, y2, z2;
    cin >> x2 >> y2 >> z2;
    ll ans = rast(x1, y1, z1, x2, y2, z2);
    ll n2 = n * 2;
    ans = min(ans, rast(x1, y1, z1, x2, y2, z2));

    ans = min(ans, rast(x1 + n2, y1 - n, z1 - n, x2, y2, z2));
    ans = min(ans, rast(x1 - n2, y1 + n, z1 + n, x2, y2, z2));

    ans = min(ans, rast(x1 + n, y1 - n2, z1 + n, x2, y2, z2));
    ans = min(ans, rast(x1 - n, y1 + n2, z1 - n, x2, y2, z2));

    ans = min(ans, rast(x1 + n, y1 + n, z1 - n2, x2, y2, z2));
    ans = min(ans, rast(x1 - n, y1 - n, z1 + n2, x2, y2, z2));

    cout << ans;
}

signed main()
{
    ios_base::sync_with_stdio(0);
    cin.tie(0);
    cin.tie(0);

    int t = 1;
    // cin >> t;

    while (t--)
    {
        solve();
    }
    return 0;
}
```

Task C

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

using namespace std;

typedef long long ll;
typedef long double ld;

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

struct Hash
{
    const ll p1 = 37;
    const ll p2 = 31;

    const ll MOD1 = 1e9 + 7;
    const ll MOD2 = 1e9 + 9;

    ll fast_mod(ll x)
    {
        x = (x >> 31) & (x & ((1ll << 31) - 1));
        x = (x >> 31) & (x & ((1ll << 31) - 1));
        return x;
    }

    vector <ll> pp1;
    vector <ll> pp2;
    vector <ll> h1;
    vector <ll> h2;

    Hash(string& s)
    {
        pp1.resize(s.size() + 1);
        pp2.resize(s.size() + 1);
        h1.resize(s.size() + 1);
        h2.resize(s.size() + 1);

        pp1[0] = 1;
        pp2[0] = 1;
        for (int i = 1; i <= s.size(); i++)
        {
            pp1[i] = pp1[i - 1] * p1 % MOD1;
            pp2[i] = pp2[i - 1] * p2 % MOD2;
        }

        for (int i = 1; i <= s.size(); i++)
        {
            h1[i] = ((h1[i - 1] * p1 % MOD1) + (s[i - 1] - '0' + 1)) %
    MOD1;
```

```
            h2[i] = ((h2[i - 1] * p2 % MOD2) + (s[i - 1] - '0' + 1)) %
    MOD2;
        }
    }

    ll get_hsh1(int l, int r)
    {
        r++;
        return ((h1[r] - h1[l] * pp1[r - l] % MOD1) + MOD1 * 2ll) % MOD1;
    }

    ll get_hsh2(int l, int r)
    {
        r++;
        return ((h2[r] - h2[l] * pp2[r - l] % MOD2) + MOD2 * 2ll) % MOD2;
    }
};

void solve()
{
    string tp;
    cin >> tp;
    int n;
    cin >> n;
    string s;
    cin >> s;
    if (tp == "first")
    {
        Hash h = Hash(s);
        cout << h.get_hsh1(0, n - 1) << ' ' << h.get_hsh2(0, n - 1);
    }
    else
    {
        ll h1, h2;
        cin >> h1 >> h2;
        s += s;
        Hash h = Hash(s);
        for (int i = 0; i < n; i++)
        {
            if (h1 == h.get_hsh1(i, i + n - 1) && h2 == h.get_hsh2(i, i +
     n - 1))
            {
                cout << "YES";
                return;
            }
        }
        cout << "NO";
    }
}

/**
first
6
```

```
011001
*/
/**
second
6
100101
73194993 30536769
*/

signed main()
{
    ios_base::sync_with_stdio(0);
    cin.tie(0);
    cin.tie(0);

    int t = 1;
    // cin >> t;

    while (t--)
    {
        solve();
    }
    return 0;
}
```

Task D

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

using namespace std;

typedef long long ll;
typedef long double ld;

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

deque <int> indx;

int req(int i, int j)
{
    cout << i + 1 << ' ' << j + 1 << endl;
    int tmp = indx.front();
    indx.pop_front();
    indx.push_back(tmp);
    char ch;
    cin >> ch;
    if (ch == '!')
    {
        exit(0);
    }
```

```
    if (ch == '=')
    {
        return 0;
    }
    if (ch == '>')
    {
        return 1;
    }
    return -1;
}

void solve()
{
    int n;
    cin >> n;
    for (int i = 0; i < n; i++)
    {
        indx.push_back(i);
    }
    while (true)
    {
        int ind1 = indx[0];
        int ind2 = indx[1];
        int ans = req(0, 0);
        if (ind2 != 0)
        {
            if (ans == 1)
            {
                req(0, n - 1);
            }
        }
    }
}

signed main()
{
    ios_base::sync_with_stdio(0);
    cin.tie(0);
    cin.tie(0);

    int t = 1;
    // cin >> t;

    while (t--)
    {
        solve();
    }
    return 0;
}
```

Task E

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

using namespace std;

typedef long long ll;
typedef long double ld;

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

mt19937 rnd = mt19937();

void dfs(int v, int p, vector <deque<int>>& vc, vector<int>& tin, vector<
    int>& sz, vector<int>& ps, int& t)
{
    tin[v] = t++;
    sz[v] = 1;
    ps[v] = p;
    for (auto u: vc[v])
    {
        if (p == u) continue;
        dfs(u, v, vc, tin, sz, ps, t);
        sz[v] += sz[u];
    }
}

void dfs2(int v, int p, vector <deque<int>>& vc, vector<int>& tin, vector
    <int>& sz, vector<int>& ps, int& t)
{
    tin[v] = t++;
    sz[v] = 1;
    ps[v] = p;
    for (auto u: vc[v])
    {
        if (p == u) continue;
        dfs(u, v, vc, tin, sz, ps, t);
        sz[v] += sz[u];
    }
}

void solve()
{
    int n, q;
    cin >> n >> q;
    vector <deque<int>> vc(n);
    //vector <deque<int>> vc2(n);
    vector <int> check(n);
    vector <int> sz(n);
    vector <int> ps(n);
    vector <int> tin(n);
    int timer = 1;
```

```
//vector <int> check2(n);
//vector <int> check2(n);
for (int i = 0; i < n; i++)
{
    ll k;
    cin >> k;
    for (int j = 0; j < k; j++)
    {
        int a;
        cin >> a;
        a--;
        vc[i].push_back(a);
        //cout << "LOL " << j << endl;
    }
    //vc2[i] = vc[i];
    //cout << i << endl;
}
dfs(0, -1, vc, tin, sz, ps, timer);
//check2 = check;
int flag = n;
vector <int> flags(n);
vector <int> path;
int t;
cin >> t;
int ind = 0;
//vector <int> path2;
vector <int> anses;
int v = 0;
while (true)
{
    if (flag == 0 && v == 0) break;
    if (!flags[v])
    {
        flag--;
    }
    path.push_back(v);
    if (path.size() - 1 == t)
    {
        anses.push_back(path[path.size() - 1]);
    }
    flags[v] = true;
    check[v]++;
    check[v] %= vc[v].size();
    int u = vc[v][0];
    vc[v].pop_front();
    vc[v].push_back(u);

    v = u;
}
/*v = 0;
while (true)
{
    if (path2.size() > 1000)
```

```
        {
            break;
        }
        path2.push_back(v);
        check2[v]++;
        check2[v] %= vc2[v].size();
        int u = vc2[v][0];
        vc2[v].pop_front();
        vc2[v].push_back(u);
        v = u;
    }*/
    v = 0;
    int ln1 = path.size();
    flag = vc[v].size();
    while (true)
    {
        if (v == 0)
        {
            flag--;
        }
        if (flag < 0)
        {
            break;
        }
        path.push_back(v);
        check[v]++;
        check[v] %= vc[v].size();
        int u = vc[v][0];
        vc[v].pop_front();
        vc[v].push_back(u);
        v = u;
    }

    int ln2 = path.size() - ln1;
        int t;
        cin >> t;
        if (t < ln1)
        {
            cout << path[t] + 1 << ' ';
        }
        else
        {
            cout << path[((t - ln1) % ln2) + ln1] + 1 << ' ';
        }
        // cout << "000" << endl;
    }
    /*cout << endl << ln1 << ' ' << ln2 << ' ' << path.size() << endl;
    for (auto a: path)
    {
        cout << a << ' ';
    }

    for (int i = 0; i < 1000; i++)
```

```
    {
        ll tt;
        int t = i;
        if (t < ln1)
        {
            tt = path[t];
        }
        else
        {
            tt = path[((t - ln1) % ln2) + ln1];
        }
        if (tt != path2[i])
        {
            cout << "OOOO: " << i << ' ' << tt << ' ' << path2[i] << endl
    ;
        }
    }*/
}

/**
6 9
2  3 2
2  4 1
3  1 5 6
1  2
1  3
1  3
0 1 2 3 4 5 6 7 8
*/

/// 1 3 1 2 4 2 1 3 5

signed main()
{
    ios_base::sync_with_stdio(0);
    cin.tie(0);
    cin.tie(0);

    int t = 1;
    // cin >> t;

    while (t--)
    {
        solve();
    }
    return 0;
}
```

Task F

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

using namespace std;
```

```
typedef long long ll;
typedef long double ld;

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

mt19937 rnd = mt19937(13379);


vector <vector<int>> ans;
vector <string> vc;
int h, w;

int gt(int i, int j)
{
    return ans[(i + h * h) % h][(j + w * w) % w];
}

void st(int i, int j, int val)
{
    ans[(i + h * h) % h][(j + w * w) % w] = val;
}

int gt_v(int i, int j)
{
    return vc[(i + h * h) % h][(j + w * w) % w] - '0';
}

int c = 1;

void rot(int i, int j)
{
    set <int> stt;
    stt.insert(gt(i, j));
    stt.insert(gt(i, j + 1));
    stt.insert(gt(i + 1, j));
    stt.insert(gt(i + 1, j + 1));
    if (stt.size() != 2) return;
    int tp1 = *stt.begin();
    int tp2 = *stt.rbegin();
    if (tp2 == gt(i, j))
    {
        swap(tp1, tp2);
    }
    if (gt(i, j) == gt(i + 1, j))
    {
        st(i, j + 1, tp1);
        st(i + 1, j, tp2);
    }
    else
```

```
    {
        st(i + 1, j, tp1);
        st(i, j + 1, tp2);
    }
}

ll get_sum()
{
    vector <bool> used(c);
    ll sum = 0;
    for (int i = 0; i < h; i++)
    {
        for (int j = 0; j < w; j++)
        {
            if (used[gt(i, j)])
            {
                continue;
            }
            ll t = 0;
            if (gt(i, j) == gt(i + 1, j))
            {
                t = gt_v(i, j) * gt_v(i + 1, j);
            }
            else if (gt(i, j) == gt(i, j + 1))
            {
                t = gt_v(i, j) * gt_v(i, j + 1);
            }
            if (t > 0)
            {
                used[gt(i, j)] = true;
                // cout << i << ' ' << j << ' ' << t << endl;
            }
            sum += t;
        }
    }
    // cout << "_____END_____\n" << endl;
    return sum;
}

void solve()
{
    cin >> h >> w;
    vc.resize(h);
    for (auto& i: vc) cin >> i;
    vector <vector<char>> tmp(h, vector<char>(w, '.'));
    ans.resize(h, vector<int>(w, 0));
    for (int i = 0; i < h; i++)
    {
        for (int j = 0; j < w; j += 2)
        {
            ans[i][j] = c;
            ans[i][j + 1] = c++;
        }
```

```
}

for (int i = 0; i < h; i++)
{
    for (int j = 0; j < w; j++)
    {
        if (gt(i, j) == gt(i + 1, j))
        {
            tmp[i][j] = 'T';
            tmp[(i + 1) % h][j] = 'B';
        }
        else if (gt(i, j) == gt(i, j + 1))
        {
            tmp[i][j] = 'L';
            tmp[i][(j + 1) % w] = 'R';
        }
    }
}
// rot(0, 0);
ld t = 0.5;
ld d_t = 0.7;
int ind1 = 0, ind2 = 0;
for (int ii = 0; ii < 10000; ii++)
{
    ll a = get_sum();
    rot(ind1, ind2);
    ll b = get_sum();
    // rot(ind1, ind2);
    if (b <= a)
    {
        int val = rnd() * t;
        if (val == 0)
        {
            rot(ind1, ind2);
        }
    }
    ind2++;
    if (ind2 == w)
    {
        ind2 = 0;
        ind1++;
    }
    ind1 %= h;
    t *= d_t;
}

// rot(0, 0);
for (int i = 0; i < h; i++)
{
    for (int j = 0; j < w; j++)
    {
        if (gt(i, j) == gt(i + 1, j))
        {
```

```
                tmp[i][j] = 'T';
                tmp[(i + 1) % h][j] = 'B';
            }
            else if (gt(i, j) == gt(i, j + 1))
            {
                tmp[i][j] = 'L';
                tmp[i][(j + 1) % w] = 'R';
            }
        }
    }
    cout << get_sum() << '\n';
    for (auto& i: tmp)
    {
        for (auto j: i)
        {
            cout << j;
        }
        cout << '\n';
    }
}

signed main()
{
    ios_base::sync_with_stdio(0);
    cin.tie(0);
    cin.tie(0);

    int t = 1;
    // cin >> t;

    while (t--)
    {
        solve();
    }
    return 0;
}
```