Олимпиада СПбГУ
по информатике
2023/24 учебного года

24fin0297 – Медведев Михаил Сергеевич

A	B	C	D	E	F	sum
100	100	100	100	20	51	471

Task A

```
//#pragma GCC optimize("Ofast,unroll-loops")
//#pragma GCC target("avx2,tune=native")
//#pragma GCC optimize("O3, unroll-loops")
//#pragma GCC target("avx2")

#include <vector>
#include <iostream>
#include <string>
#include <map>
#include <algorithm>
#include <set>
#include <queue>
#include <cmath>
#include <stack>
#include <bitset>
#include <unordered_map>
#include <random>
#include <chrono>

#define int long long
#define ull __int128
typedef long long ll;
typedef long double ld;

using namespace std;


void solve() {
    int a, b, n;
```

```
    cin >> a >> b >> n;
    int z = max({n * 2, a + 1, b + 1});
    vector<int> dp(z);
    int ans = 0;
    dp[a] = 1;
    for (int i = a + 1; i <= n; i++) {
        dp[i] = dp[i - a];
        if (i >= b) dp[i] += dp[i - b];
    }
    ans += dp[n] * a;
    dp.assign(z, 0);
    dp[b] = 1;
    for (int i = b + 1; i <= n; i++) {
        dp[i] = dp[i - b];
        if (i >= a) dp[i] += dp[i - a];
    }
    ans += dp[n] * b;
    cout << ans;

}

signed main() {
    ios_base::sync_with_stdio(false);
    cin.tie(nullptr);
    cout.tie(nullptr);

    int t = 1;
//    cin >> t;
    while (t) {
        t--;
        solve();
    }
}
```

Task B

```
//#pragma GCC optimize("Ofast,unroll-loops")
//#pragma GCC target("avx2,tune=native")
//#pragma GCC optimize("O3, unroll-loops")
//#pragma GCC target("avx2")

#include <vector>
#include <iostream>
#include <string>
#include <map>
#include <algorithm>
#include <set>
#include <queue>
#include <cmath>
#include <stack>
#include <bitset>
#include <unordered_map>
#include <random>
```

```
#include <chrono>

#define int long long
#define ull __int128
typedef long long ll;
typedef long double ld;

using namespace std;



void solve() {
    int n;
    cin >> n;
    int x1, y1, z1;
    cin >> x1 >> y1 >> z1;
    int x2, y2, z2;
    cin >> x2 >> y2 >> z2;
    int ans = max({abs(x1 - x2), abs(y1 - y2), abs(z1 - z2)});
    vector<tuple<int, int, int>> v={{-n, 2 *n, -n}, {n, -2 * n, n}, {2 *n
   , -n, -n}, {-2 *n, n, n}, {-n, -n, 2 * n}, {n, n, -2 * n}};
    for (auto [a, b, c] : v) {
        x1 += a;
        y1 += b;
        z1 += c;
        ans = min(ans, max({abs(x1 - x2), abs(y1 - y2), abs(z1 - z2)}));
        x1 -= a;
        y1 -= b;
        z1 -= c;
    }
    cout << ans;
}

signed main() {

    ios_base::sync_with_stdio(false);
    cin.tie(nullptr);
    cout.tie(nullptr);


    int t = 1;
//    cin >> t;
    while (t) {
        t--;
        solve();
    }
}

//
```

Task C

```
//#pragma GCC optimize("Ofast,unroll-loops")
```

```
//#pragma GCC target("avx2,tune=native")
//#pragma GCC optimize("O3, unroll-loops")
//#pragma GCC target("avx2")

#include <vector>
#include <iostream>
#include <string>
#include <map>
#include <algorithm>
#include <set>
#include <queue>
#include <cmath>
#include <stack>
#include <bitset>
#include <unordered_map>
#include <random>
#include <chrono>

#define int long long
#define ull __int128
typedef long long ll;
typedef long double ld;

using namespace std;

mt19937 mt1(1329);

vector<int> k={151, 239, 7, 257, 13};
vector<int> p={(int) 1e9 + 7, (int) 1e9 + 9, (int) 1e9 + 7, (int) 1e9 +
    9, (int) 1e9 + 7};

void solve() {
    int a, b, n;
    cin >> a >> b >> n;
    int z = max({n * 2, a + 1, b + 1});
    vector<int> dp(z);
    int ans = 0;
    dp[a] = 1;
    for (int i = a + 1; i <= n; i++) {
        dp[i] = dp[i - a];
        if (i >= b) dp[i] += dp[i - b];
    }
    ans += dp[n] * a;
    dp.assign(z, 0);
    dp[b] = 1;
    for (int i = b + 1; i <= n; i++) {
        dp[i] = dp[i - b];
        if (i >= a) dp[i] += dp[i - a];
    }
    ans += dp[n] * b;
    cout << ans;

}
```

```
void first() {
    int n;
    cin >> n;
    string s;
    cin >> s;
    for (int j = 0; j < 5; j++) {
        int res = 0;
        for (int i = 0; i < n; i++) {
            res = (res * k[j] + (s[i] - '0')) % p[j];
        }
        cout << res << " ";
    }
    cout << endl;
}

void set_prefs(string& s, vector<int>& prefs, int j) {
    prefs.assign(s.size() + 1, 0);
    for (int i = 1; i < s.size() + 1; i++) {
        prefs[i] = (prefs[i - 1] * k[j] + (s[i - 1] - '0')) % p[j];
    }
}

void set_pows(string& s, vector<int>& pows, int j) {
    pows.push_back(1);
    for (int i = 0; i < s.size(); i++) {
        pows.push_back((pows[i] * k[j]) % p[j]);
    }
}

int sub_hash(int a, int b, vector<int>& prefs, vector<int>& pows, int j)
    {
    int h = (prefs[b + 1] - (prefs[a] * pows[b - a + 1]) % p[j] + p[j]) %
     p[j];
    return h;
}

void second() {
    int n;
    cin >> n;
    string s, t;
    cin >> s;
    vector<int> h(5);
    for (int &i : h) {
        cin >> i;
    }

    s += s;
    vector<vector<int>> pows(5), prefs(5);
    for (int j = 0; j < 5; j++) {
        set_prefs(s, prefs[j], j);
        set_pows(s, pows[j], j);
```

```
    }

    for (int i = 0; i < n; i++) {
        bool f = true;
        for (int j = 0; j < 5; j++) {
            if (sub_hash(i, i + n - 1, prefs[j], pows[j], j) != h[j]) {
                f = false;
                break;
            }
        }
        if (f) {
            cout << "YES";
            cout << endl;
            return;
        }
    }
    cout << "NO";
    cout << endl;

}


signed main() {

    ios_base::sync_with_stdio(false);
    cin.tie(nullptr);
    cout.tie(nullptr);

    string type;
    cin >> type;
    if (type == "first") {
        first();
    } else {
        second();
    }

//    int t = 1;
////    cin >> t;
//    while (t) {
//        t--;
//        solve();
//    }
}

//
```

Task D

```
//#pragma GCC optimize("Ofast,unroll-loops")
//#pragma GCC target("avx2,tune=native")
//#pragma GCC optimize("O3, unroll-loops")
//#pragma GCC target("avx2")
```

```cpp
#include <vector>
#include <iostream>
#include <string>
#include <map>
#include <algorithm>
#include <set>
#include <queue>
#include <cmath>
#include <stack>
#include <bitset>
#include <unordered_map>
#include <random>
#include <chrono>

#define int long long
#define ull __int128
typedef long long ll;
typedef long double ld;

using namespace std;

mt19937 rnd1(1329);

int ask(int i, int j) {
    cout << i + 1 << " " << j + 1 << endl;
    char res;
    cin >> res;
    if (res == '!') exit(0);
    if (res == '>') return -1;
    return 1;
}

void solve() {
    int n;
    cin >> n;
    deque<int> inds(n);
    for (int i = 0; i < n; i++) {
        inds[i] = i;
    }
    vector<vector<int>> checks(n, vector<int> (n));
    int cnt = n * (n - 1) / 2;
    while (cnt) {
        int s = -1;
        for (int i = 1; i < n; i++) {
            if (!checks[inds[0]][inds[i]]) {
                s = i;
                break;
            }
        }
        if (s == -1) {
            ask(0, 0);
        } else {
            int res = ask(1, s);
```

```
            swap(inds[1], inds[s]);
            cnt--;
            checks[inds[0]][inds[1]] = res;
            checks[inds[1]][inds[0]] = -res;
        }

        inds.push_back(inds[0]);
        inds.pop_front();
    }
    deque<int> v(n, 1);
    for (int i = 0; i < n; i++) {
        int ind = inds[i];
        for (int j = 0; j < n; j++) {
            if (checks[ind][j] == 1) {
                v[i]++;
            }
        }
    }

//    for (int i : v) {
//        cout << i << " ";
//    }
//    cout << endl;

    while (true) {
        if (v[0] < v[1] && !(v[0] == 1 && v[1] == n)) {
            ask(0, 1);
            swap(v[0], v[1]);
        } else {
            ask(0, 0);
        }
        v.push_back(v[0]);
        v.pop_front();
    }
}

signed main() {

    ios_base::sync_with_stdio(false);
    cin.tie(nullptr);
    cout.tie(nullptr);


    int t = 1;
//    cin >> t;
    while (t) {
        t--;
        solve();
    }
}
```

Task E

```
//#pragma GCC optimize("Ofast,unroll-loops")
//#pragma GCC target("avx2,tune=native")
//#pragma GCC optimize("O3, unroll-loops")
//#pragma GCC target("avx2")

#include <vector>
#include <iostream>
#include <string>
#include <map>
#include <algorithm>
#include <set>
#include <queue>
#include <cmath>
#include <stack>
#include <bitset>
#include <unordered_map>
#include <random>
#include <chrono>

#define int long long
#define ull __int128
typedef long long ll;
typedef long double ld;

using namespace std;

vector<vector<int>> g;
vector<int> used, p, ques, ans;
int cnt = 0, t = 0, now_q = 0;

void dfs(int v) {
    if (t == ques[now_q]) {
        ans[now_q] = v;
        now_q++;
    }
    if (now_q == ques.size()) return;
    t++;
    used[v] = 1;
    cnt--;
    while (true) {
        int go = g[v][p[v] % g[v].size()];
        p[v]++;
        dfs(go);

        if (now_q == ques.size()) return;
    }
}

void solve() {
    int n, q;
    cin >> n >> q;
    g.resize(n);
    used.resize(n);
```

```
    p.resize(n);
    cnt = n;
    for (int i = 0; i < n; i++) {
        int d;
        cin >> d;
        for (int j = 0; j < d; j++) {
            int x;
            cin >> x;
            x--;
            g[i].push_back(x);
        }
    }
    ques.resize(q);

    ans.resize(q);
    for (int &i : ques) {
        cin >> i;
    }
    dfs(0);
    for (int i : ans) {
        cout << i + 1 << " ";
    }
}

signed main() {

    ios_base::sync_with_stdio(false);
    cin.tie(nullptr);
    cout.tie(nullptr);


    int tt = 1;
//    cin >> tt;
    while (tt) {
        tt--;
        solve();
    }
}
```

Task F

```
//#pragma GCC optimize("Ofast,unroll-loops")
//#pragma GCC target("avx2,tune=native")
//#pragma GCC optimize("O3, unroll-loops")
//#pragma GCC target("avx2")

#include <vector>
#include <iostream>
#include <string>
#include <map>
#include <algorithm>
#include <set>
#include <queue>
```

```
#include <cmath>
#include <stack>
#include <bitset>
#include <unordered_map>
#include <random>
#include <chrono>

#define int long long
#define ull __int128
typedef long long ll;
typedef long double ld;

using namespace std;
int h, w;
int x(int x) {
    return (x + h) % h;
}
int y (int y) {
    return (y + w) % w;
}

mt19937 rnd1(1329);

int get(int x, vector<string> &v) {
    int a = x / w;
    int b = x % w;
    return v[a][b] - '0';
}

void solve() {

    cin >> h >> w;
    vector<string> v(h);
    for (int i = 0; i < h; i++) {
        cin >> v[i];
    }
    vector<vector<pair<int, int>>> g(h * w);
    for (int i = 0; i < h; i++) {
        for (int j = 0; j < w; j++) {
            g[i * w + j].emplace_back(x(i - 1) * w + j, 1);
            g[i * w + j].emplace_back(x(i + 1) * w + j, 2);
            g[i * w + j].emplace_back(i * w + y(j - 1), 3);
            g[i * w + j].emplace_back(i * w + y(j + 1), 4);
        }
    }
    int n = h * w;
    int best = 0;
    vector<int> ans;
    for (int t = 0; t < 500; t++) {
        vector<int> used(h * w), d(h * w, -1);
        for (int i = 0; i < h * w; i++) {
            if (used[i]) continue;
            int x = rnd1() % g[i].size();
```

```
            for (int j = x; j < g[i].size(); j++) {
                if (!used[g[i][j].first]) {
                    used[g[i][j].first] = 1;
                    used[i] = 1;
                    d[i] = j;
                    break;
                }
            }
        }
        int res = 0;
        for (int i = 0; i < n; i++) {
            if (d[i] != -1) {
               res += get(i, v) * get(g[i][d[i]].first, v);
            }
        }
        if (res > best) {
            best = res;
            ans = d;
        }
    }
    cout << best;
    vector<char> out(n, '.');
    for (int i = 0; i < n; i++) {
        if (ans[i] == -1) continue;
        int type = g[i][ans[i]].second;
        int j = g[i][ans[i]].first;
        if (type == 1) {
            out[i] = 'B';
            out[j] = 'T';
        } else if (type == 2) {
            out[i] = 'T';
            out[j] = 'B';
        }  else if (type == 3) {
            out[i] = 'R';
            out[j] = 'L';
        }  else if (type == 4) {
            out[i] = 'L';
            out[j] = 'R';
        }
    }

    for (int i = 0; i < n; i++) {
        if (i % w == 0) cout << endl;
        cout << out[i];

    }
}

signed main() {

    ios_base::sync_with_stdio(false);
    cin.tie(nullptr);
    cout.tie(nullptr);
```

```
    int tt = 1;
//    cin >> tt;
    while (tt) {
        tt--;
        solve();
    }
}
```