Олимпиада СПбГУ
по информатике
2023/24 учебного года

24fin0331 – Новгородцев Григорий Алексеевич

A	B	C	D	E	F	sum
100	100	100	100	40	90	530

Task A

```
#include <bits/stdc++.h>
#define int long long

using namespace std;
using ll = long long;


signed main() {
    ios_base::sync_with_stdio(false); cin.tie(0); cout.tie(0);
#ifdef LOCAL
    freopen("input.txt", "r", stdin);
    freopen("output.txt", "w", stdout);
#endif
    int a, b, n; cin >> a >> b >> n;
    int ans = 0;
    vector<int> dp(n + 1);
    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];
    }
    if (n >= a)
        ans = dp[n - a] * a;
    if (n >= b)
        ans += dp[n - b] * b;
    cout << ans;
    return 0;
}
```

Task B

```
#include <bits/stdc++.h>
#define int long long

using namespace std;
using ll = long long;


signed main() {
    ios_base::sync_with_stdio(false); cin.tie(0); cout.tie(0);
#ifdef LOCAL
    freopen("input.txt", "r", stdin);
    freopen("output.txt", "w", stdout);
#endif
    int n; cin >> n;
    int x1, y1, z1; cin >> x1 >> y1 >> z1;
    int x2, y2, z2; cin >> x2 >> y2 >> z2;
    int ans = (abs(x1 - x2) + abs(y1 - y2) + abs(z1 - z2)) / 2;
    vector<int> tmp = {x1 - x2, y1 - y2, z1 - z2};
    for (int coor = 0; coor < 3; ++coor) {
        for (int mn = -1; mn <= 1; mn += 2) {
            auto tmp2 = tmp;
            for (int i = 0; i < 3; ++i) {
                if (i == coor) {
                    tmp2[i] += 2 * n * mn;
                } else {
                    tmp2[i] += n * mn * -1;
                }
            }
            ans = min(ans, (abs(tmp2[0]) + abs(tmp2[1]) + abs(tmp2[2])) /
 2);
        }
    }
    cout << ans;
    return 0;
}
```

Task C

```
#include <bits/stdc++.h>
#define int long long

using namespace std;
using ll = long long;
const int MOD = 1e9 + 7;
const int b = 5;
const int MAXN = 3e5 + 5;

int st[MAXN + 5];

struct HashStr {
```

```
    string s;
    vector<int> h;

    HashStr(string _s) {
        s = _s;
        h.resize(s.size() + 1);
        h[0] = 0;
        for (int i = 1; i <= (int)s.size(); ++i) {
            h[i] = (h[i - 1] + (st[i] * (s[i - 1] - '0' + 1) % MOD) % MOD
   );
        }
    }

    int getSub(int l, int r) { // seg 0 numer
        r++;
        return (st[MAXN - r] * ((h[r] - h[l] + MOD) % MOD) % MOD);
    }
};

void solveFirst() {
    int n; cin >> n;
    string s; cin >> s;
    HashStr t(s);
    cout << t.getSub(0, n - 1) << '\n';
}

void solveSecond() {
    int n; cin >> n;
    string s; cin >> s;
    s += s;
    HashStr t(s);
    int need; cin >> need;
    for (int i = 0; i < n; ++i) {
        if (need == t.getSub(i, i + n - 1)) {
            cout << "YES\n";
            return;
        }
    }
    cout << "NO\n";
}

signed main() {
    ios_base::sync_with_stdio(false); cin.tie(0); cout.tie(0);
#ifdef LOCAL
    freopen("input.txt", "r", stdin);
    freopen("output.txt", "w", stdout);
#endif
    st[0] = 1;
    for (int i = 1; i < MAXN + 5; ++i) st[i] = (st[i - 1] * b) % MOD;
    string r; cin >> r;
    if (r == "first") {
        solveFirst();
    } else {
```

```
        solveSecond();
    }
    return 0;
}
```

Task D

```
#include <bits/stdc++.h>
#define int long long

using namespace std;
using ll = long long;

signed main() {
    ios_base::sync_with_stdio(false); cin.tie(0); cout.tie(0);
#ifdef LOCAL
//    freopen("input.txt", "r", stdin);
//    freopen("output.txt", "w", stdout);
#endif
    int n; cin >> n;
    vector<vector<int>> g(n, vector<int>(n, -1));
    vector<int> p(n);
    iota(p.begin(), p.end(), 0);
    int cnt = 0;
    for (int i = 0;; ++i) {
        if (cnt >= ((n * (n - 1)) / 2)) break;
        int ind = -1;
        for (int j = 1; j < n; ++j) {
            if (g[p[0]][p[j]] == -1) {
                ind = j;
                break;
            }
        }
//        if (ind == -1) break;
        if (ind == 0 || ind == -1) {
            cout << "1 1" << endl;
            char ans; cin >> ans;
            if (ans == '!') return 0;
            rotate(p.begin(), p.begin() + 1, p.end());
            continue;
        }
        cnt++;
        cout << 2 << ' ' << ind + 1 << endl;
        swap(p[1], p[ind]);
        char ans; cin >> ans;
        if (ans == '!') return 0;
        if (ans == '>') {
            g[p[0]][p[1]] = 0;
            g[p[1]][p[0]] = 1;
        } else {
            g[p[0]][p[1]] = 1;
            g[p[1]][p[0]] = 0;
        }
```

```
        rotate(p.begin(), p.begin() + 1, p.end());
    }
    vector<int> num(n, 1);
    for (int i = 0; i < n; ++i) {
        for (int j = 0; j < n; ++j) {
            num[i] += (g[i][j] == 1);
        }
    }
    vector<int> now(n);
    for (int i = 0; i < n; ++i) now[i] = num[p[i]];
    for (int i = 0; i < n - 1; ++i) {
        int ind = 0;
        if (now[0] != n - i) {
            for (int j = 1; j < n; ++j) {
                if (now[j] == n - i) {
                    ind = j;
                    break;
                }
            }
        }
        swap(now[0], now[ind]);
        cout << 1 << ' ' << ind + 1 << endl;
        char ans; cin >> ans;
        if (ans == '!') return 0;
        rotate(now.begin(), now.begin() + 1, now.end());
    }
    cout << "1 1" << endl;
    char ans; cin >> ans;
    return 0;
}
```

Task E

```
#include <bits/stdc++.h>
#define int long long

using namespace std;
using ll = long long;

vector<vector<int>> g;
vector<int> ptr;
vector<int> euler;

void dfs(int v, int p) {
    euler.push_back(v);
    for (int i = ptr[v]; g[v][i] != p;) {
        dfs(g[v][i], v);
        euler.push_back(v);
        i++;
        if (i == (int)g[v].size()) i = 0;
        if (v == 0 && ptr[v] == i) break;
    }
}
```

```
signed main() {
    ios_base::sync_with_stdio(false); cin.tie(0); cout.tie(0);
#ifdef LOCAL
    freopen("input.txt", "r", stdin);
    freopen("output.txt", "w", stdout);
#endif
    int n, q; cin >> n >> q;
    g.resize(n);
    for (int i = 0; i < n; ++i) {
        int k; cin >> k;
        g[i].resize(k);
        for (int j = 0; j < k; ++j) {
            cin >> g[i][j];
            g[i][j]--;
        }
    }
    vector<int> t(q);
    for (int i = 0; i < q; ++i) cin >> t[i];
    ptr.resize(n);
    vector<int> used(n);
    int cntUnused = n - 1;
    int v = 0;
    used[0] = 1;
    vector<int> path = {0};
    while (cntUnused || v != 0) {
        if (path.size() > t.back() + 3) break;
        int u = g[v][ptr[v]];
        ptr[v]++;
        if (ptr[v] == (int) g[v].size()) ptr[v] = 0;
        v = u;
        if (!used[v]) {
            used[v] = 1;
            cntUnused--;
        }
        path.push_back(v);
    }
    path.pop_back();
    dfs(0, 0);
//    for (auto i : euler) cerr << i + 1 << ' ';
    euler.pop_back();
    for (int i = 0; i < q; ++i) {
        if (t[i] < path.size()) {
            cout << path[t[i]] + 1 << '\n';
        } else {
            t[i] -= path.size();
            t[i] %= euler.size();
            cout << euler[t[i]] + 1 << '\n';
        }
    }
    return 0;
}
```

Task F

```
#include <bits/stdc++.h>
#define int long long

using namespace std;
using ll = long long;

const vector<pair<int, int>> DELTA = {{0, 1}, {0, -1}, {1, 0}, {-1, 0}};

int n, m;

pair<int, int> makeNorm(int i, int j) {
    if (i == -1) i = n - 1;
    if (i == n) i = 0;
    if (j == -1) j = m - 1;
    if (j == m) j = 0;
    return {i, j};
}

signed main() {
    ios_base::sync_with_stdio(false); cin.tie(0); cout.tie(0);
#ifdef LOCAL
    freopen("input.txt", "r", stdin);
    freopen("output.txt", "w", stdout);
#endif
    cin >> n >> m;
    vector<vector<int>> a(n, vector<int>(m));
    for (int i = 0; i < n; ++i) {
        for (int j = 0; j < m; ++j) {
            char c; cin >> c;
            a[i][j] = c - '0';
        }
    }
    vector<vector<char>> ans(n, vector<char>(m, '.'));
    int numAns = 0;
    while (true) {
        int best = -1;
        pair<int, int> best1, best2;
        bool hor = false;
        for (int i = 0; i < n; ++i) {
            for (int j = 0; j < m; ++j) {
                for (auto [di, dj] : DELTA) {
                    int ni = i + di;
                    int nj = j + dj;
                    auto [i2, j2] = makeNorm(ni, nj);
                    if (ans[i][j] == '.' && ans[i2][j2] == '.') {
                        int now = a[i][j] * a[i2][j2];
                        if (now > best) {
                            if (di == 0) hor = true;
                            else hor = false;
                            best1 = {i, j};
                            best2 = {i2, j2};
```

```
                        best = now;
                    }
                }
            }
        }
    }
    if (best == -1) break;
    numAns += best;
    if (hor) {
        if (abs(best1.second - best2.second) == 1) {
            if (best2.second < best1.second) swap(best2, best1);
        } else {
            if (best1.second < best2.second) swap(best2, best1);
        }
        ans[best1.first][best1.second] = 'L';
        ans[best2.first][best2.second] = 'R';
    } else {
        if (abs(best1.first - best2.first) == 1) {
            if (best1.first > best2.first) swap(best1, best2);
        } else {
            if (best2.first > best1.first) swap(best1, best2);
        }
        ans[best1.first][best1.second] = 'T';
        ans[best2.first][best2.second] = 'B';
    }
}
cout << numAns << '\n';
for (int i = 0; i < n; ++i) {
    for (auto j : ans[i]) {
        cout << j;
    }
    cout << '\n';
}
return 0;
}
```