Олимпиада СПбГУ
по информатике
2023/24 учебного года

24fin0072 – Бушева Милана Максимовна

A	B	C	D	E	F	sum
100	100	100	100	40	50	490

Task A

```
a, b, n = map(int, input().split())
facts = [1, 1]
for i in range(2, n + 1):
    facts.append(facts[i - 1] * i)
x = 0
ans = 0
while (a * x <= n):
    if (n - (a * x)) % b == 0:
        y = (n - (a * x)) // b
        ans += n * (facts[x + y] // facts[x] // facts[y]) // (x + y)
    x += 1
print(ans)
```

Task B

```
#include <iostream>
#include <vector>
#include <cstdlib>
#include <cstdio>
#include <cmath>
#include <algorithm>
#include <numeric>
#include <tuple>
#include <set>
#include <map>
#include <array>

using namespace std;
```

```
using ull = unsigned long long;
using ll = long long;

ll get_dist(array<ll, 3>& a, array<ll, 3>& b) {
    ll ans = 0;
    for (int i = 0; i < 3; i++) {
        ans += abs(a[i] - b[i]);
    }
    return ans / 2;
}

int main() {
    cin.tie(0)->sync_with_stdio(false);

    int n;
    cin >> n;
    array<ll, 3> a, b;
    for (int i = 0; i < 3; i++) cin >> a[i];
    for (int j = 0; j < 3; j++) cin >> b[j];

    ll ans = get_dist(a, b);
    for (int i = 0; i >= -n; i--) { //x
        int j = -i - n;
        array<ll, 3> f = { n, i, j }, s = { -n, -j, -i };
        ans = min(ans, get_dist(a, f) + get_dist(b, s));
        ans = min(ans, get_dist(a, s) + get_dist(b, f));
    }
    for (int i = 0; i >= -n; i--) { //y
        int j = -i - n;
        array<ll, 3> f = { i, n, j }, s = { -j, -n, -i };
        ans = min(ans, get_dist(a, f) + get_dist(b, s));
        ans = min(ans, get_dist(a, s) + get_dist(b, f));
    }
    for (int i = 0; i >= -n; i--) { //z
        int j = -i - n;
        array<ll, 3> f = { i, j, n }, s = { -j, -i, -n };
        ans = min(ans, get_dist(a, f) + get_dist(b, s));
        ans = min(ans, get_dist(a, s) + get_dist(b, f));
    }
    cout << ans;

    return 0;
}
```

Task C

```
#include <iostream>
#include <vector>
#include <cstdlib>
#include <cstdio>
#include <cmath>
#include <algorithm>
#include <numeric>
```

```
#include <tuple>
#include <set>
#include <map>
#include <array>
#include <string>

using namespace std;

using ull = unsigned long long;
using ll = long long;


const int mod = 1e9 + 9, p = 53;
const int SIZE = 1e5 + 1;
int powers[SIZE];

int add(int a, int b) {
    return a + b >= mod ? a + b - mod : a + b;
}

int rem(int a, int b) {
    return a - b < 0 ? a - b + mod : a - b;
}

int mul(int a, int b) {
    return 1ll * a * b % mod;
}

int get_hash(string& s) {
    int n = s.size();
    int h = 0;
    for (int i = 0; i < n; i++) {
        h = add(mul(h, p), s[i] - '0' + 1);
    }
    return h;
}

void fst() {
    int n;
    cin >> n;
    string s;
    cin >> s;
    int h = get_hash(s);
    cout << h << endl;
}

void scd() {
    int n;
    cin >> n;
    string s;
    cin >> s;
    string h;
    cin >> h;
```

```
    int hs = stoi(h);
    powers[0] = 1;
    for (int i = 1; i < SIZE; i++) {
        powers[i] = mul(powers[i - 1], p);
    }
    int myh = get_hash(s);
    for (int i = 0; i < n; i++) {
        if (hs == myh) {
            cout << "YES\n";
            exit(0);
        }
        myh = rem(myh, mul(s[i] - '0' + 1, powers[n - 1]));
        myh = add(mul(myh, p), s[i] - '0' + 1);
    }
    if (hs == myh) {
        cout << "YES\n";
    } else {
        cout << "NO\n";
    }
}

int main() {
    cin.tie(0)->sync_with_stdio(false);

    string s;
    cin >> s;
    if (s == "first") fst();
    else scd();


    return 0;
}
```

Task D

```
#include <iostream>
#include <vector>
#include <cstdlib>
#include <cstdio>
#include <cmath>
#include <algorithm>
#include <numeric>
#include <tuple>
#include <set>
#include <map>
#include <array>
#include <string>

using namespace std;

using ull = unsigned long long;
using ll = long long;
```

```
int q = 0;

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

int main() {
    cin.tie(0)->sync_with_stdio(false);

    int n;
    cin >> n;
    char now = '<';
    while (now != '!') {
        for (int i = 0; i < n - 1; i++) {
            if (now == '>') {
                now = ask(1, n);
            } else {
                now = ask(1, 1);
            }
        }
        now = ask(1, 1);
    }


    return 0;
}
```

Task E

```
#include <iostream>
#include <vector>
#include <cstdlib>
#include <cstdio>
#include <cmath>
#include <algorithm>
#include <numeric>
#include <tuple>
#include <set>
#include <map>
#include <array>
#include <string>

using namespace std;

using ull = unsigned long long;
using ll = long long;
```

```
int n, q;
ll mx;


void dfs1(vector<vector<int>>& gr, vector<int>&num, vector<int>&res, int
    v, int& now) {
    res[now++] = v;
    if (now > mx) return;
    int k = num[v];
    num[v] = (num[v] + 1) % gr[v].size();
    dfs1(gr, num, res, gr[v][k], now);
}

void solve1(vector<vector<int>>& gr, vector<ll>&t) {
    vector<int>num(n), res(1e5 + 2);
    int now = 0;
    mx = t.back();
    dfs1(gr, num, res, 0, now);
    for (ll i : t) {
        cout << res[i] + 1 << ' ';
    }
}

void dfs2(vector<vector<int>>& gr, vector<int>& num, vector<int>& res,
    vector<bool>&used, int v) {
    if (v == 0 && count(used.begin(), used.end(), false) == 0) {
        return;
    }
    res.push_back(v);
    used[v] = true;

    int k = num[v];
    num[v] = (num[v] + 1) % gr[v].size();
    dfs2(gr, num, res, used, gr[v][k]);
}

void dfs3(vector<vector<int>>& gr,vector<int>&num,  int v, vector<int>&
    ans) {
    ans.push_back(v);
    int end = gr[v].size() - 1;
    if (v == 0) end++;
    for (int i = 0; i < end; i++) {
        dfs3(gr, num, gr[v][num[v]], ans);
        ans.push_back(v);
        num[v] = (num[v] + 1) % gr[v].size();
    }
}

void solve2(vector<vector<int>>& gr, vector<ll>& t) {
    vector<int>num(n), res;
    vector<bool>used(n);
    dfs2(gr, num, res, used, 0);
```

```
    int tm = res.size() - 1;
    vector<int> ans;
    dfs3(gr, num, 0, ans);
    ans.pop_back();
    for (ll i : t) {
        if (i <= tm) {
            cout << res[i] + 1 << ' ';
        } else {
            cout << ans[(i - tm - 1) % ans.size()] + 1 << ' ';
        }
    }
}

int main() {
    cin.tie(0)->sync_with_stdio(false);

    cin >> n >> q;
    vector<vector<int>> gr(n);
    for (int i = 0; i < n; i++) {
        int k;
        cin >> k;
        while (k--) {
            int v;
            cin >> v;
            gr[i].push_back(v - 1);
        }
    }
    vector<ll> t(q);
    for (int i = 0; i < q; i++) {
        cin >> t[i];
    }
    if (n <= 1000) {
        solve2(gr, t);
    }
    else if (t.back() <= 1e5) {
        solve1(gr, t);
    }

    return 0;
}
```

Task F

```
#include <iostream>
#include <vector>
#include <cstdlib>
#include <cstdio>
#include <cmath>
#include <algorithm>
#include <numeric>
#include <tuple>
#include <set>
#include <map>
```

```
#include <array>
#include <string>

using namespace std;

using ull = unsigned long long;
using ll = long long;

int h, w;
vector<string> vec;

ll var1() {
    ll ans = 0;
    for (int i = 0; i < h; i++) {
        for (int j = 0; j < w; j += 2) {
            ans += (vec[i][j] - '0') * (vec[i][j + 1] - '0');
        }
    }
    return ans;
}

ll var2() {
    ll ans = 0;
    for (int i = 0; i < h; i++) {
        ans += (vec[i][0] - '0') * (vec[i].back() - '0');
        for (int j = 1; j < w - 1; j += 2) {
            ans += (vec[i][j] - '0') * (vec[i][j + 1] - '0');
        }
    }
    return ans;
}

ll var3() {
    ll ans = 0;
    for (int i = 0; i < w; i++) {
        for (int j = 0; j < h; j += 2) {
            ans += (vec[j][i] - '0') * (vec[j + 1][i] - '0');
        }
    }
    return ans;
}

ll var4() {
    ll ans = 0;
    for (int i = 0; i < w; i++) {
        ans += (vec[0][i] - '0') * (vec.back()[i] - '0');
        for (int j = 1; j < h - 1; j += 2) {
            ans += (vec[j][i] - '0') * (vec[j + 1][i] - '0');
        }
    }
    return ans;
}
```

```
ll var5() {
    ll ans = 0;

    for (int i = 0; i < h; i+=2) {
        for (int j = 0; j < w; j += 2) {
            ans += max((vec[i][j] - '0') * (vec[i][j + 1] - '0') + (vec[i
     + 1][j] - '0') * (vec[i + 1][j + 1] - '0'),
                (vec[i][j] - '0') * (vec[i + 1][j] - '0') + (vec[i][j +
    1] - '0') * (vec[i + 1][j + 1] - '0'));
        }
    }
    return ans;
}

ll var6() {
    ll ans = 0;

    for (int i = 1; i < h - 1; i += 2) {
        for (int j = 0; j < w; j += 2) {
            ans += max((vec[i][j] - '0') * (vec[i][j + 1] - '0') + (vec[i
     + 1][j] - '0') * (vec[i + 1][j + 1] - '0'),
                (vec[i][j] - '0') * (vec[i + 1][j] - '0') + (vec[i][j +
    1] - '0') * (vec[i + 1][j + 1] - '0'));
        }
    }
    for (int j = 0; j < w; j += 2) {
        ans += max((vec[0][j] - '0') * (vec[0][j + 1] - '0') + (vec.back
    ()[j] - '0') * (vec.back()[j + 1] - '0'),
            (vec[0][j] - '0') * (vec.back()[j] - '0') + (vec[0][j + 1] -
    '0') * (vec.back()[j + 1] - '0'));
    }
    return ans;
}

void genvar1() {
    string s = "";
    for (int i = 0; i < w; i += 2) {
        s += "LR";
    }
    for (int i = 0; i < h; i++) {
        cout << s << '\n';
    }
}

void genvar2() {
    string s = "";
    for (int i = 0; i < w; i += 2) {
        s += "RL";
    }
    for (int i = 0; i < h; i++) {
        cout << s << '\n';
    }
}
```

```
void genvar3() {
    string s1 = string(w, 'T'), s2 = string(w, 'B');
    for (int i = 0; i < h; i++) {
        if (i % 2 == 0) {
            cout << s1 << '\n';
        } else {
            cout << s2 << '\n';
        }
    }
}

void genvar4() {
    string s1 = string(w, 'T'), s2 = string(w, 'B');
    for (int i = 0; i < h; i++) {
        if (i % 2 == 1) {
            cout << s1 << '\n';
        } else {
            cout << s2 << '\n';
        }
    }
}

void genvar5() {
    vector<string> res(h);
    for (int i = 0; i < h; i += 2) {
        for (int j = 0; j < w; j += 2) {
            ll a = (vec[i][j] - '0') * (vec[i][j + 1] - '0') + (vec[i +
    1][j] - '0') * (vec[i + 1][j + 1] - '0');
            ll b = (vec[i][j] - '0') * (vec[i + 1][j] - '0') + (vec[i][j
    + 1] - '0') * (vec[i + 1][j + 1] - '0');
            if (a > b) {
                res[i] += "LR";
                res[i + 1] += "LR";
            } else {
                res[i] += "TT";
                res[i + 1] += "BB";
            }
        }
    }
    for (int i = 0; i < h; i++) {
        cout << res[i] << '\n';
    }
}

void genvar6() {
    vector<string> res(h);
    for (int i = 1; i < h - 1; i += 2) {
        for (int j = 0; j < w; j += 2) {
            ll a = (vec[i][j] - '0') * (vec[i][j + 1] - '0') + (vec[i +
    1][j] - '0') * (vec[i + 1][j + 1] - '0');
            ll b = (vec[i][j] - '0') * (vec[i + 1][j] - '0') + (vec[i][j
    + 1] - '0') * (vec[i + 1][j + 1] - '0');
```

```
            if (a > b) {
                res[i] += "LR";
                res[i + 1] += "LR";
            } else {
                res[i] += "TT";
                res[i + 1] += "BB";
            }
        }
    }
    for (int j = 0; j < w; j += 2) {
        ll a = (vec[0][j] - '0') * (vec[0][j + 1] - '0') + (vec.back()[j]
 - '0') * (vec.back()[j + 1] - '0');
        ll b = (vec[0][j] - '0') * (vec.back()[j] - '0') + (vec[0][j + 1]
 - '0') * (vec.back()[j + 1] - '0');
        if (a > b) {
            res[0] += "LR";
            res.back() += "LR";
        } else {
            res[0] += "BB";
            res.back() += "TT";
        }
    }
    for (int i = 0; i < h; i++) {
        cout << res[i] << '\n';
    }
}


int main() {
    cin.tie(0)->sync_with_stdio(false);

    cin >> h >> w;
    vec.resize(h);
    for (int i = 0; i < h; i++) {
        cin >> vec[i];
    }
    ll ans1 = var1(), ans2 = var2(), ans3 = var3(), ans4 = var4(), ans5 =
 var5(), ans6 = var6();
    ll mx = max({ ans1, ans2, ans3, ans4 });
    cout << mx << '\n';
    if (mx == ans1) {
        genvar1();
    } else if (mx == ans2) {
        genvar2();
    } else if (mx == ans3) {
        genvar3();
    } else if (mx == ans4) {
        genvar4();
    } else if (mx == ans5){
        genvar5();
    } else {
        genvar6();
    }
```

```
    return 0;
}
```