Олимпиада СПбГУ
по информатике
2023/24 учебного года

24fin0321 – Наседкин Дмитрий Сергеевич

A	B	C	D	E	F	sum
100	100	100	35	20	91	446

Task A

```
#include <iostream>
#include <algorithm>
#include <set>
#include <unordered_set>
#include <map>
#include <unordered_map>
#include <queue>
#include <math.h>
#include <vector>
#include <string>
#include <stack>
#include <cassert>
#include <random>
#include <chrono>
#include <cstring>
#include <iomanip>
using namespace std;

#define all(_x) _x.begin(), _x.end()
#define rall(_x) _x.rbegin(), _x.rend()
#define sz(_x) (int)_x.size()
#define ff first
#define ss second
#define pb push_back

using ll = long long;
using ull = unsigned long long;
using vi = vector<int>;
using vll = vector<ll>;
```

```
using pii = pair<int, int>;
using pll = pair <ll, ll>;
using vpii = vector <pii>;
using vvi = vector <vi>;

const int INF = 1e9;
const ll INFLL = 1e18;
const int mod = 1e9 + 7;
const double eps = 1e-8;

int a, b;
const int MAXN = 80;
ll dp[MAXN + 1];

ll get_ans(int n) {
    if (n < 0)
        return 0;
    if (n == 0)
        return 1;
    if (dp[n] == -1)
        dp[n] = get_ans(n - a) + get_ans(n - b);
    return dp[n];
}

void solve() {
    memset(dp, -1, sizeof(ll) * (MAXN + 1));
    int n;
    cin >> a >> b >> n;
    get_ans(n - a);
    get_ans(n - b);
    ll res1 = get_ans(n - a) * a;
    ll res2 = get_ans(n - b) * b;
    cout << res1 + res2 << '\n';
}

int main() {
    ios_base::sync_with_stdio(0);
    cin.tie(0);
    solve();
}
```

Task B

```
#include <iostream>
#include <algorithm>
#include <set>
#include <unordered_set>
#include <map>
#include <unordered_map>
#include <queue>
#include <math.h>
#include <vector>
#include <string>
```

```
#include <stack>
#include <cassert>
#include <random>
#include <chrono>
#include <cstring>
#include <iomanip>
using namespace std;

#define all(_x) _x.begin(), _x.end()
#define rall(_x) _x.rbegin(), _x.rend()
#define sz(_x) (int)_x.size()
#define ff first
#define ss second
#define pb push_back

using ll = long long;
using ull = unsigned long long;
using vi = vector<int>;
using vll = vector<ll>;
using pii = pair<int, int>;
using pll = pair <ll, ll>;
using vpii = vector <pii>;
using vvi = vector <vi>;

const int INF = 1e9;
const ll INFLL = 1e18;
const int mod = 1e9 + 7;
const double eps = 1e-8;

int get_dist(int x1, int y1, int z1, int x2, int y2, int z2) {
    int dist_x = x2 - x1;
    int dist_y = y2 - y1;
    int dist_z = z2 - z1;
    int max_dist = max(max(abs(dist_x), abs(dist_y)), abs(dist_z));
    return max_dist;
}

void solve() {
    int n, x1, y1, z1, x2, y2, z2;
    cin >> n >> x1 >> y1 >> z1 >> x2 >> y2 >> z2;
    int res = INF;
    res = min(res, get_dist(x1, y1, z1, x2, y2, z2));
    res = min(res, get_dist(x1, y1, z1, x2 + 2 * n, y2 - n, z2 - n));
    res = min(res, get_dist(x1, y1, z1, x2 - 2 * n, y2 + n, z2 + n));
    res = min(res, get_dist(x1, y1, z1, x2 - n, y2 + 2 * n, z2 - n));
    res = min(res, get_dist(x1, y1, z1, x2 + n, y2 - 2 * n, z2 + n));
    res = min(res, get_dist(x1, y1, z1, x2 - n, y2 - n, z2 + 2 * n));
    res = min(res, get_dist(x1, y1, z1, x2 + n, y2 + n, z2 - 2 * n));
    cout << res << '\n';
}

int main() {
    ios_base::sync_with_stdio(0);
```

```
    cin.tie(0);
    solve();
}
```

Task C

```
#include <iostream>
#include <algorithm>
#include <set>
#include <unordered_set>
#include <map>
#include <unordered_map>
#include <queue>
#include <math.h>
#include <vector>
#include <string>
#include <stack>
#include <cassert>
#include <random>
#include <chrono>
#include <cstring>
#include <iomanip>
using namespace std;

#define all(_x) _x.begin(), _x.end()
#define rall(_x) _x.rbegin(), _x.rend()
#define sz(_x) (int)_x.size()
#define ff first
#define ss second
#define pb push_back

using ll = long long;
using ull = unsigned long long;
using vi = vector<int>;
using vll = vector<ll>;
using pii = pair<int, int>;
using pll = pair <ll, ll>;
using vpii = vector <pii>;
using vvi = vector <vi>;

const int INF = 1e9;
const ll INFLL = 1e18;
const int mod = 1e9 + 7;
const double eps = 1e-8;

const int MAXN = 2e5;
const int power = 31;
ll p[MAXN + 1];
int p_mod[MAXN + 1];
ll pref[MAXN + 1];
ll pref_mod[MAXN + 1];

void init() {
```

```
    p[0] = 1;
    p_mod[0] = 1;
    for (int i = 1; i <= MAXN; i++) {
        p[i] = p[i - 1] * power;
        p_mod[i] = (1LL * p_mod[i - 1] * power) % mod;
    }
}

ll get_hash(string s) {
    ll res = 0;
    for (char c : s) {
        res *= power;
        res += (c - '0' + 1);
    }
    return res;
}
int get_hash_mod(string s) {
    int res = 0;
    for (char c : s)
        res = (1LL * res * power + c - '0' + 1) % mod;
    return res;
}

void get_pref(string s) {
    for (int i = 0; i < sz(s); i++)
        pref[i + 1] = pref[i] * power + (s[i] - '0' + 1);
}
void get_pref_mod(string s) {
    for (int i = 0; i < sz(s); i++)
        pref_mod[i + 1] = (1LL * pref_mod[i] * power + (s[i] - '0' + 1))
   % mod;
}

ll get_substr(int l, int r) {
    return pref[r] - pref[l] * p[r - l];
}
int get_mod(int a, int b) {
    a %= b;
    if (a < 0)
        a += b;
    return a;
}
int get_substr_mod(int l, int r) {
    return get_mod(pref_mod[r] - int((1LL * pref_mod[l] * p_mod[r - l]) %
     mod), mod);
}

void code() {
    int n;
    string s;
    cin >> n >> s;
    cout << get_hash(s) << ' ' << get_hash_mod(s);
}
```

```
void decode() {
    init();
    int n;
    string t;
    ll h;
    int h_mod;
    cin >> n >> t >> h >> h_mod;
    t += t;
    get_pref(t);
    get_pref_mod(t);
    for (int i = 0; i < n; i++) {
        //cout << get_substr(i, i + n) << ' ' << get_substr_mod(i, i + n)
     << '\n';
        if (get_substr(i, i + n) == h && get_substr_mod(i, i + n) ==
    h_mod) {
            cout << "YES\n";
            return;
        }
    }
    cout << "NO\n";
}

void solve() {
    string s;
    cin >> s;
    if (s == "first")
        code();
    else
        decode();
}

int main() {
    ios_base::sync_with_stdio(0);
    cin.tie(0);
    solve();
}
```

Task D

```
#include <iostream>
#include <algorithm>
#include <set>
#include <unordered_set>
#include <map>
#include <unordered_map>
#include <queue>
#include <math.h>
#include <vector>
#include <string>
#include <stack>
#include <cassert>
#include <random>
#include <chrono>
```

```
#include <cstring>
#include <iomanip>
using namespace std;

#define all(_x) _x.begin(), _x.end()
#define rall(_x) _x.rbegin(), _x.rend()
#define sz(_x) (int)_x.size()
#define ff first
#define ss second
#define pb push_back

using ll = long long;
using ull = unsigned long long;
using vi = vector<int>;
using vll = vector<ll>;
using pii = pair<int, int>;
using pll = pair <ll, ll>;
using vpii = vector <pii>;
using vvi = vector <vi>;

const int INF = 1e9;
const ll INFLL = 1e18;
const int mod = 1e9 + 7;
const double eps = 1e-8;

char ask(int i, int j) {
    char res;
    cout << i << ' ' << j << endl;
    cin >> res;
    return res;
}

void solve() {
    int n;
    cin >> n;
    while (true) {
        int i = rand() % n;
        int j = rand() % n;
        if (i == j)
            j = (j + 1) % n;
        char res = ask(i + 1, j + 1);
        if (res == '!')
            break;
        for (int k = 0; k < n; k++) {
            res = ask(1, 1);
            if (res == '!')
                return;
        }
    }
}

int main() {
    ios_base::sync_with_stdio(0);
```

```
    cin.tie(0);
    solve();
}
```

Task E

```
#include <iostream>
#include <algorithm>
#include <set>
#include <unordered_set>
#include <map>
#include <unordered_map>
#include <queue>
#include <math.h>
#include <vector>
#include <string>
#include <stack>
#include <cassert>
#include <random>
#include <chrono>
#include <cstring>
#include <iomanip>
using namespace std;

#define all(_x) _x.begin(), _x.end()
#define rall(_x) _x.rbegin(), _x.rend()
#define sz(_x) (int)_x.size()
#define ff first
#define ss second
#define pb push_back

using ll = long long;
using ull = unsigned long long;
using vi = vector<int>;
using vll = vector<ll>;
using pii = pair<int, int>;
using pll = pair <ll, ll>;
using vpii = vector <pii>;
using vvi = vector <vi>;

const int INF = 1e9;
const ll INFLL = 1e18;
const int mod = 1e9 + 7;
const double eps = 1e-8;

const int MAXN = 1e5;
int n;
int shift[MAXN];
vi g[MAXN];
int t = 0;
unordered_set<int> q;

void dfs(int v, int p = -1) {
```

```
    if (p != -1)
        shift[p] = (shift[p] + 1) % sz(g[p]);
    if (q.empty())
        return;
    if (q.count(t)) {
        q.erase(t);
        cout << v + 1 << ' ';
    }
    t++;
    dfs(g[v][shift[v]], v);
}

void solve() {
    int m;
    cin >> n >> m;
    for (int i = 0; i < n; i++) {
        int k;
        cin >> k;
        g[i].resize(k);
        for (int &u : g[i]) {
            cin >> u;
            u--;
        }
    }
    for (int i = 0; i < m; i++) {
        int x;
        cin >> x;
        q.insert(x);
    }
    dfs(0);
}

int main() {
    ios_base::sync_with_stdio(0);
    cin.tie(0);
    solve();
}
```

Task F

```
#include <iostream>
#include <algorithm>
#include <set>
#include <unordered_set>
#include <map>
#include <unordered_map>
#include <queue>
#include <math.h>
#include <vector>
#include <string>
#include <stack>
#include <cassert>
#include <random>
```

```
#include <chrono>
#include <cstring>
#include <iomanip>
using namespace std;

#define all(_x) _x.begin(), _x.end()
#define rall(_x) _x.rbegin(), _x.rend()
#define sz(_x) (int)_x.size()
#define ff first
#define ss second
#define pb push_back

using ll = long long;
using ull = unsigned long long;
using vi = vector<int>;
using vll = vector<ll>;
using pii = pair<int, int>;
using pll = pair <ll, ll>;
using vpii = vector <pii>;
using vvi = vector <vi>;

const int INF = 1e9;
const ll INFLL = 1e18;
const int mod = 1e9 + 7;
const double eps = 1e-8;

const int MAXN = 50;
int h, w;
int a[MAXN][MAXN];
int used[MAXN][MAXN];
vpii best;

void init() {
    for (int i = 0; i < 10; i++)
        for (int j = 0; j < 10; j++)
            best.pb({i, j});
    sort(rall(best), [&](pii a, pii b) {
        return a.ff * a.ss < b.ff * b.ss;
    });
}

void solve() {
    init();
    cin >> h >> w;
    for (int i = 0; i < h; i++) {
        string s;
        cin >> s;
        for (int j = 0; j < w; j++)
            a[i][j] = (s[j] - '0');
    }

    int ans = 0;
    for (auto [s, q] : best) {
```

```
        for (int i = 0; i < h; i++)
            for (int j = 0; j < w; j++) {
                if (!used[i][j] && !used[i][(j + 1) % w] && a[i][j] == s
    && a[i][(j + 1) % w] == q) {
                    used[i][j] = 1;
                    used[i][(j + 1) % w] = 2;
                    ans += s * q;
                }
                if (!used[i][j] && !used[(i + 1) % h][j] && a[i][j] == s
    && a[(i + 1) % h][j] == q) {
                    used[i][j] = 3;
                    used[(i + 1) % h][j] = 4;
                    ans += s * q;
                }
            }
    }

    cout << ans << '\n';
    for (int i = 0; i < h; i++) {
        for (int j = 0; j < w; j++) {
            if (used[i][j] == 1)
                cout << 'L';
            else if (used[i][j] == 2)
                cout << 'R';
            else if (used[i][j] == 3)
                cout << 'T';
            else if (used[i][j] == 4)
                cout << 'B';
            else
                cout << '.';
        }
        cout << '\n';
    }
}

int main() {
    ios_base::sync_with_stdio(0);
    cin.tie(0);
    solve();
}
```