Олимпиада СПбГУ
по информатике
2023/24 учебного года

24fin0243 – Кудряшов Алексей Андреевич

A	B	C	D	E	F	sum
100	100	100	75	20	92	487

Task A

```
#include <bits/stdc++.h>
using namespace std;
using ll = long long;
const int N = 200;
ll dp[N];
signed main() {
    cin.tie(nullptr)->sync_with_stdio(false);
    ll a, b, n;
    cin >> a >> b >> n;
    if (n < a) {
        cout << 0ll << "\n";
        return 0;
    }
    dp[0] = 1ll;
    for (int i = 1; i <= n; i++) dp[i] = (i >= a ? dp[i - a] : 0ll) + (i
    >= b ? dp[i - b] : 0ll);
    ll ans = dp[n - a] * a + (n >= b ? dp[n - b] * b : 0);
    cout << ans << "\n";
}
```

Task B

```
#include <bits/stdc++.h>
using namespace std;
using ll = long long;
struct Point{
    ll x, y, z;
    Point() {
        x = y = z = 0;
```

```
    }
    Point (ll xx, ll yy, ll zz) {
        x = xx;
        y = yy;
        z = zz;
    }
};
ll dist(Point a, Point b) {
    return max({abs(a.y - b.y), abs(a.z - b.z), abs(a.x - b.x)});
}
Point invx(Point a) {
    return Point(-a.x, -a.z, -a.y);
}
Point invy(Point a) {
    return Point(-a.z, -a.y, -a.x);
}
Point invz(Point a) {
    return Point(-a.y, -a.x, -a.z);
}
void chmin(ll &a, ll b) {
    a = min(a, b);
}
signed main() {
    cin.tie(nullptr)->sync_with_stdio(false);
    int n;
    cin >> n;
    Point st, fn;
    cin >> st.x >> st.y >> st.z;
    cin >> fn.x >> fn.y >> fn.z;
    ll ANS = dist(st, fn);
    // fixed X
    for (int y = 0; y >= -n; y--) {
        Point p = Point(n, y, - n - y);
        Point invp = invx(p);
        chmin(ANS, dist(p, st) + dist(invp, fn));
        chmin(ANS, dist(invp, st) + dist(p, fn));
    }
    // fixed Y
    for (int x = 0; x >= -n; x--) {
        Point p = Point(x, n, - n - x);
        Point invp = invy(p);
        chmin(ANS, dist(p, st) + dist(invp, fn));
        chmin(ANS, dist(invp, st) + dist(p, fn));
    }
    // fixed Z
    for (int x = 0; x >= -n; x--) {
        Point p = Point(x, - n - x, n);
        Point invp = invz(p);
        chmin(ANS, dist(p, st) + dist(invp, fn));
        chmin(ANS, dist(invp, st) + dist(p, fn));
    }
    cout << ANS << "\n";
}
```

Task C

```
#include <bits/stdc++.h>
using namespace std;
using ll = long long;
const ll p = 37;
const ll MOD = 1e9 + 9;
void first() {
    int n;
    cin >> n;
    string s;
    cin >> s;
    ll H = 0;
    for (int i = 0; i < n; i++) {
        H = H * p % MOD + s[i];
    }
    cout << bitset<100>(H).to_string() << "\n";
}
void second() {
    int n;
    cin >> n;
    string t;
    cin >> t;
    t.append(t);
    string h;
    cin >> h;
    ll RES = bitset<100>(h).to_ullong();
    ll H = 0;
    ll P = 1;
    for (int i = 0; i < n - 1; i++) P = P * p % MOD;
    for (int i = 0; i < n; i++) H = H * p % MOD + t[i];
    for (int i = n; i < 2 * n; i++) {
        if (H == RES) {
            cout << "YES\n";
            return ;
        }
        H = ((H - P * t[i-n]) % MOD + MOD) % MOD;
        H = H * p % MOD + t[i];
    }
    cout << "NO\n";
}
signed main() {
    cin.tie(nullptr)->sync_with_stdio(false);
    string s;
    cin >> s;
    if (s == "first") {
        first();
    } else {
        second();
    }
}
```

Task D

```
#include <bits/stdc++.h>
using namespace std;
using ll = long long;
int n;
int start = 0;
int interact(int p1, int p2) {
    int t1 = (start + p1) % n;
    int t2 = (start + p2) % n;
    cout << t1 + 1 << " " << t2 + 1 << endl;
    char ch;
    cin >> ch;
    start--;
    if (start == -1) start += n;
    if (ch == '!') exit(0);
    if (ch == '<') return -1;
    if (ch == '>') return 1;
    return 0;
}
bool ask(int a, int b) { // ? a < b
    if (a == 0 && b == 1) {
        while (start != 0) interact(0, 0);
        auto res = interact(0, 0);
        return res == 1;
    }
    if (a == 1 && b == 0) {
        while (start != 0) interact(0, 0);
        auto res = interact(0, 0);
        return res == -1;
    }
    interact(0, a);
    interact(1, b);
    while (start != 0) interact(0, 0);
    int res = interact(0, 0);
    interact(1, b);
    interact(0, a);
    return res == 1;
}
signed main() {
    cin.tie(nullptr)->sync_with_stdio(false);
    cin >> n;
    for (int i = 0; i < n; i++) {
        for (int j = i + 1; j < n; j++) {
            bool res = ask(i, j);
            if (res) {
                interact(i, j);
            }
        }
    }
    for (int i = 0; i < n; i++) interact(0, 0);
    assert(false);
}
```

Task E

```
#include <bits/stdc++.h>
using namespace std;
using ll = long long;
const int N = 101010;
int ans[N];
signed main() {
    cin.tie(nullptr)->sync_with_stdio(false);
    int n, q;
    cin >> n >> q;
    vector<deque<int>> g(n);
    for (int i = 0; i < n; i++) {
        int sz;
        cin >> sz;
        for (int j = 0; j < sz; j++) {
            int x;
            cin >> x;
            x--;
            g[i].push_back(x);
        }
    }
    int v = 0;
    for (int i = 0; i < N; i++) {
        ans[i] = v + 1;
        int t = g[v].front();
        g[v].pop_front();
        g[v].push_back(t);
        v = t;
    }
    for (int i = 0; i < q; i++) {
        int x;
        cin >> x;
        cout << ans[x] << "\n";
    }
}
```

Task F

```
#include <bits/stdc++.h>
using namespace std;
using ll = long long;
mt19937 rnd(1337);
long double rnd2() {
    return 1.0 * rnd() / rnd.max();
}
const int N = 51;
int a[N][N];
int CUR = 0;
int TOP = 0;
char out[N][N];
char t[N][N];
```

```
int h, w;
void ch(int x, int y) {
    int x1 = (x - 1 + h) % h;
    int y1 = (y - 1 + w) % w;
    if (t[x][y] == 'R') {
        CUR -= a[x][y] * a[x][y1] + a[x1][y] * a[x1][y1];
        t[x][y] = 'B';
        t[x][y1] = 'B';
        t[x1][y] = 'T';
        t[x1][y1] = 'T';
        CUR += a[x][y] * a[x1][y] + a[x][y1] * a[x1][y1];
    } else {
        CUR -= a[x][y] * a[x1][y] + a[x][y1] * a[x1][y1];
        t[x][y] = 'R';
        t[x][y1] = 'L';
        t[x1][y] = 'R';
        t[x1][y1] = 'L';
        CUR += a[x][y] * a[x][y1] + a[x1][y] * a[x1][y1];
    }
}
vector<vector<int>> order;
int dx[] = {0, 0, -1, 1};
int dy[] = {-1, 1, 0, 0};
signed main() {
    cin.tie(nullptr)->sync_with_stdio(false);
    cin >> h >> w;
    for (int i = 0; i < h; i++) {
        for (int j = 0; j < w; j++) {
            char ch;
            cin >> ch;
            a[i][j] = ch - '0';
            t[i][j] = '.';
        }
    }
    for (int i = 0; i < h; i++) {
        for (int j = 0; j < w; j++) {
            for (int dir = 0; dir < 4; dir++) {
                int newx = (i + dx[dir] + h) % h;
                int newy = (j + dy[dir] + w) % w;
                order.push_back({a[i][j] * a[newx][newy], i, j, dir});
            }
        }
    }
    sort(order.rbegin(), order.rend());
    for (int i = 0; i < order.size(); i++) {
        int x = order[i][1];
        int y = order[i][2];
        int dir = order[i][3];
        int newx = (x + dx[dir] + h) % h;
        int newy = (y + dy[dir] + w) % w;
        int s = order[i][0];
        if (t[x][y] == '.') {
            if (dir == 0 && t[newx][newy] == '.') {
```

```
                CUR += s;
                t[x][y] = 'R';
                t[x][newy] = 'L';
            } else if(dir == 1 && t[newx][newy] == '.') {
                CUR += s;
                t[x][y] = 'L';
                t[x][newy] = 'R';
            } else if(dir == 2 && t[newx][newy] == '.') {
                CUR += s;
                t[x][y] = 'B';
                t[newx][y] = 'T';
            } else if(dir == 3 && t[newx][newy] == '.') {
                CUR += s;
                t[x][y] = 'T';
                t[newx][y] = 'B';
            }
        }
    }

    TOP = CUR;
    for (int i = 0; i < h; i++) for (int j = 0; j < w; j++) out[i][j] = t
    [i][j];

    int N = 4e7;

    for (int iter = 0; iter < N; iter++) {
        int x = rnd() % h;
        int y = rnd() % w;
        int x1 = (x - 1 + h) % h;
        int y1 = (y - 1 + w) % w;
        if (t[x][y] == 'R' && t[x1][y] == 'R' || t[x][y] == 'B' && t[x][
    y1] == 'B') {
            int last = CUR;
            ch(x, y);
            if (CUR < last && rnd2() > exp(-1.0 * iter / N * 0.5 * (last
    - CUR))) ch(x, y);

            if (CUR > TOP) {
                TOP = CUR;
                for (int i = 0; i < h; i++) for (int j = 0; j < w; j++)
    out[i][j] = t[i][j];
            }
        }
    }
    cout << TOP << "\n";
    for (int i = 0; i < h; i++) {
        for (int j = 0; j < w; j++) cout << out[i][j];
        cout << "\n";
    }
}
```