Олимпиада СПбГУ
по информатике
2023/24 учебного года

24fin0452 – Толмачев Александр Александрович

A	B	C	D	E	F	sum
100	100	100	75	20	67	462

Task A

```
#include <bits/stdc++.h>
#include <ext/pb_ds/assoc_container.hpp>
using namespace std;
using namespace __gnu_pbds;
#define ordered_set tree<int, null_type, less<int>, rb_tree_tag,
    tree_order_statistics_node_update>
const int INF = 1e10;
const int maxn = 2e5;
mt19937_64 gen(time(nullptr));
#define int long long
int ansf = 0, anss = 0;
int solve(int a, int b, int n) {
    if (n < 0) {
        return 0;
    }
    if (n == 0) {
        return 1;
    }
    int f = solve(a, b, n-a);
    int s = solve(a, b, n-b);
    ansf = f;
    anss = s;
    return f+s;
}
signed main() {
    ios_base::sync_with_stdio(false);
    cin.tie(nullptr);
    int a, b, n;
    cin >> a >> b >> n;
```

```
    vector<pair<int, int>> dp(n+1000);
    dp[0].first = 1;
    for (int i = 0; i < n; i++) {
        dp[i+a].first += dp[i].first+dp[i].second;
        dp[i+b].second += dp[i].first+dp[i].second;
    }
    cout << dp[n].first*a +dp[n].second*b;
}
```

Task B

```
#include <bits/stdc++.h>
#include <ext/pb_ds/assoc_container.hpp>
using namespace std;
using namespace __gnu_pbds;
#define ordered_set tree<int, null_type, less<int>, rb_tree_tag,
    tree_order_statistics_node_update>
#define int long long
const int INF = 1e10;
const int maxn = 2e5;
mt19937_64 gen(time(nullptr));

signed main() {
    ios_base::sync_with_stdio(false);
    cin.tie(nullptr);
    int n;
    cin >> n;
    int xa, ya, za;
    cin >> xa >> ya >> za;
    int xb, yb, zb;
    cin >> xb >> yb >> zb;
    int xaa = xa, yaa = ya, zaa = za;
    int ans1 = abs(xb-xa);
    xa = xb;
    ya = 0 - xa - za;
    ans1 += abs(ya-yb);
    ya = yb;
    za = 0 - xa-ya;
    ans1 += abs(za-zb);
    xa = xaa;
    ya = yaa;
    za = zaa;
    int ans2 = abs(yb-ya);
    ya = yb;
    za = 0 - xa - ya;
    ans2 += abs(za-zb);
    za = zb;
    xa = 0 - za-ya;
    ans2 += abs(xa-xb);
    xa = xaa;
    ya = yaa;
    za = zaa;
    int ans3 = abs(zb-za);
```

```
za = zb;
xa = 0 - za - ya;
ans3 += abs(xa-xb);
xa = xb;
ya = 0 - za-xa;
ans3 += abs(ya-yb);
//cout << min(ans1, min(ans2, ans3));

xa = xaa;
ya = yaa;
za = zaa;
int xbb = xb, ybb = yb, zbb = zb;
xb = n+n-abs(xb);
yb = yb-n;
zb = zb-n;
int ans4 = abs(xb-xa);
xa = xb;
ya = 0 - xa - za;
ans4 += abs(ya-yb);
ya = yb;
za = 0 - xa-ya;
ans4 += abs(za-zb);
xa = xaa;
ya = yaa;
za = zaa;
int ans5 = abs(yb-ya);
ya = yb;
za = 0 - xa - ya;
ans5 += abs(za-zb);
za = zb;
xa = 0 - za-ya;
ans5 += abs(xa-xb);
xa = xaa;
ya = yaa;
za = zaa;
int ans6 = abs(zb-za);
za = zb;
xa = 0 - za - ya;
ans6 += abs(xa-xb);
xa = xb;
ya = 0 - za-xa;
ans6 += abs(ya-yb);
xb = xbb;
yb = ybb;
zb = zbb;
xa = xaa;
ya = yaa;
za = zaa;

yb = n+n-abs(yb);
xb = xb-n;
zb = zb-n;
int ans7 = abs(xb-xa);
```

```
xa = xb;
ya = 0 - xa - za;
ans7 += abs(ya-yb);
ya = yb;
za = 0 - xa-ya;
ans7 += abs(za-zb);
xa = xaa;
ya = yaa;
za = zaa;
int ans8 = abs(yb-ya);
ya = yb;
za = 0 - xa - ya;
ans8 += abs(za-zb);
za = zb;
xa = 0 - za-ya;
ans8 += abs(xa-xb);
xa = xaa;
ya = yaa;
za = zaa;
int ans9 = abs(zb-za);
za = zb;
xa = 0 - za - ya;
ans9 += abs(xa-xb);
xa = xb;
ya = 0 - za-xa;
ans9 += abs(ya-yb);
xb = xbb;
yb = ybb;
zb = zbb;
xa = xaa;
ya = yaa;
za = zaa;
zb = n+n-abs(zb);
xb = xb-n;
yb = yb-n;
int ans10 = abs(xb-xa);
xa = xb;
ya = 0 - xa - za;
ans10 += abs(ya-yb);
ya = yb;
za = 0 - xa-ya;
ans10 += abs(za-zb);
xa = xaa;
ya = yaa;
za = zaa;
int ans11 = abs(yb-ya);
ya = yb;
za = 0 - xa - ya;
ans11 += abs(za-zb);
za = zb;
xa = 0 - za-ya;
ans11 += abs(xa-xb);
xa = xaa;
```

```
ya = yaa;
za = zaa;
int ans12 = abs(zb-za);
za = zb;
xa = 0 - za - ya;
ans12 += abs(xa-xb);
xa = xb;
ya = 0 - za-xa;
ans12 += abs(ya-yb);
xb = xbb;
yb = ybb;
zb = zbb;

xa = xaa;
ya = yaa;
za = zaa;
xb = -n-(n-abs(xb));
yb = n+yb;
zb = n+zb;
int ans13 = abs(xb-xa);
xa = xb;
ya = 0 - xa - za;
ans13 += abs(ya-yb);
ya = yb;
za = 0 - xa-ya;
ans13 += abs(za-zb);
xa = xaa;
ya = yaa;
za = zaa;
int ans14 = abs(yb-ya);
ya = yb;
za = 0 - xa - ya;
ans14 += abs(za-zb);
za = zb;
xa = 0 - za-ya;
ans14 += abs(xa-xb);
xa = xaa;
ya = yaa;
za = zaa;
int ans15 = abs(zb-za);
za = zb;
xa = 0 - za - ya;
ans15 += abs(xa-xb);
xa = xb;
ya = 0 - za-xa;
ans15 += abs(ya-yb);
xb = xbb;
yb = ybb;
zb = zbb;
xa = xaa;
ya = yaa;
za = zaa;
yb = -n-(n-abs(yb));
```

```
xb+=n;
zb+=n;
int ans16 = abs(xb-xa);
xa = xb;
ya = 0 - xa - za;
ans16 += abs(ya-yb);
ya = yb;
za = 0 - xa-ya;
ans16 += abs(za-zb);
xa = xaa;
ya = yaa;
za = zaa;
int ans17 = abs(yb-ya);
ya = yb;
za = 0 - xa - ya;
ans17 += abs(za-zb);
za = zb;
xa = 0 - za-ya;
ans17 += abs(xa-xb);
xa = xaa;
ya = yaa;
za = zaa;
int ans18 = abs(zb-za);
za = zb;
xa = 0 - za - ya;
ans18 += abs(xa-xb);
xa = xb;
ya = 0 - za-xa;
ans18 += abs(ya-yb);
xb = xbb;
yb = ybb;
zb = zbb;
xa = xaa;
ya = yaa;
za = zaa;
zb = -n-(n-abs(zb));
xb = xb+n;
yb += n;
int ans19 = abs(xb-xa);
xa = xb;
ya = 0 - xa - za;
ans19 += abs(ya-yb);
ya = yb;
za = 0 - xa-ya;
ans19 += abs(za-zb);
xa = xaa;
ya = yaa;
za = zaa;
int ans20 = abs(yb-ya);
ya = yb;
za = 0 - xa - ya;
ans20 += abs(za-zb);
za = zb;
```

```
    xa = 0 - za-ya;
    ans20 += abs(xa-xb);
    xa = xaa;
    ya = yaa;
    za = zaa;
    int ans21 = abs(zb-za);
    za = zb;
    xa = 0 - za - ya;
    ans21 += abs(xa-xb);
    xa = xb;
    ya = 0 - za-xa;
    ans21 += abs(ya-yb);
    xb = xbb;
    yb = ybb;
    zb = zbb;

    cout << min({ans1, ans2, ans3, ans4, ans5, ans6, ans7, ans8, ans9,
    ans10, ans11, ans12, ans13, ans14, ans15, ans16, ans17, ans18, ans19,
    ans20, ans21});
}
```

Task C

```
#include <bits/stdc++.h>
#include <ext/pb_ds/assoc_container.hpp>
using namespace std;
using namespace __gnu_pbds;
#define ordered_set tree<int, null_type, less<int>, rb_tree_tag,
    tree_order_statistics_node_update>
#define int long long
const int INF = 1e10;
const int p=39;
const int mod = 2286661337;
const int maxn = 3e5;
mt19937_64 gen(time(nullptr));
int get_hash(vector<int>& h, vector<int>& deg, int l, int r) {
    return (h[r+1]%mod - h[l]*deg[r-l+1] %mod +mod*mod) % mod;
}
signed main() {
    ios_base::sync_with_stdio(false);
    cin.tie(nullptr);
    string type;
    cin >> type;
    if (type[0]=='f') {
        int n;
        cin >> n;
        string s;
        cin >> s;
        vector<int> h(n+1);
        h[0] = 0;
        vector<int> deg(maxn);
        deg[0] = 1;
        for (int i = 0; i < n; i++) {
```

```
            h[i+1] = (h[i]*p%mod + s[i]-'0'+1)%mod;
        }
        for (int i = 1; i < maxn; i++) {
            deg[i] = (deg[i-1]*p)%mod;
        }
        cout << get_hash(h, deg, 0, n-1);
    } else {
        int n;
        cin >> n;
        string s;
        cin >> s;
        int hp;
        cin >> hp;
        vector<int> h(2*n+2);
        h[0] = 0;
        vector<int> deg(maxn);
        deg[0] = 1;
        for (int i = 1; i < maxn; i++) {
            deg[i] = (deg[i-1]*p)%mod;
        }
        for (int i = 0; i < 2*n; i++) {
            h[i+1] = (h[i]*p%mod + s[i%n]-'0'+1)%mod;
        }
        for (int i = 0; i < n; i++) {
            if (get_hash(h, deg,i, i+n-1)==hp) {
                cout << "YES";
                return 0;
            }
        }
        cout << "NO";
    }
}
```

Task D

```
#include <bits/stdc++.h>
#include <ext/pb_ds/assoc_container.hpp>
using namespace std;
using namespace __gnu_pbds;
#define ordered_set tree<int, null_type, less<int>, rb_tree_tag,
    tree_order_statistics_node_update>
const int INF = 1e10;
const int p=39;
const int mod = 2286661337;
const int maxn = 3e5;
mt19937_64 gen(time(nullptr));
int get_hash(vector<int>& h, vector<int>& deg, int l, int r) {
    return (h[r+1]%mod - h[l]*deg[r-l+1] %mod +mod*mod) % mod;
}
signed main() {
    ios_base::sync_with_stdio(false);
    cin.tie(nullptr);
    int n;
```

```
cin >> n;
vector<vector<int>> arr(n, vector<int>(2, 0));
for (int i = 0; i < n; i++) {
    cout << "1 1" << endl;
    char s;
    cin >> s;
    if (s == '>') {
        arr[i][1]++;
    } else if (s == '<'){
        arr[i][0]++;
    } else if (s=='!'){
        return 0;
    }
}
for (int i = 0; i < n; i++) {
    for (int j = 0; j < i; j++) {
        cout << "1 1" << endl;
        char f;
        cin >> f;
        if (f=='!') {
            return 0;
        }
    }
    for (int x = 1; x < n-1; x++) {
        cout << "2 " << 2+x << endl;
        char s;
        cin >> s;
        if (s == '>') {
            arr[i][1]++;
        } else if (s == '<'){
            arr[i][0]++;
        } else if (s == '!'){
            return 0;
        }
        cout << "1 " << 1+x << endl;
        cin >> s;
        for (int j = 0; j < n-2; j++) {
            cout << "1 1" << endl;
            char f;
            cin >> f;
            if (f=='!') {
                return 0;
            }
        }
    }
    for (int j = 0; j < n-i; j++) {
        cout << "1 1" << endl;
        char f;
        cin >> f;
        if (f=='!') {
            return 0;
        }
    }
```

```
}
vector<int> nums(n);
deque<int> d;
for (int i = 0; i < n; i++) {
    nums[i] = arr[i][0]+1;
    d.push_back(nums[i]);
}
while (true) {
    if (d.front()==n) {
        int check = 1;
        for (int i = 1; i < n; i++) {
            if (d[i] != n-i) {
                check = 0;
                int fnd = -1;
                for (int j = i+1; j < n; j++) {
                    if (d[j]==n-i) {
                        fnd = j;
                        break;
                    }
                }
                d[fnd] = d[i];
                d[i] = n-i;
                cout << i+1 << " " << fnd+1 << endl;
                char s;
                cin >> s;
                if (s == '!') {
                    return 0;
                }
                int elem = d.front();
                d.pop_front();
                d.push_back(elem);
                break;
            }
        }
        if (check == 1) {
            char t;
            cin >> t;
            if (t=='!') {
                return 0;
            }
            break;
        }
    } else {
        cout << "1 1" << endl;
        char t;
        cin >> t;
        if (t=='!') {
            return 0;
        }
        int elem = d.front();
        d.pop_front();
        d.push_back(elem);
    }
```

```
    }
}
```

Task E

```
#include <bits/stdc++.h>
#include <ext/pb_ds/assoc_container.hpp>
using namespace std;
using namespace __gnu_pbds;
#define ordered_set tree<int, null_type, less<int>, rb_tree_tag,
    tree_order_statistics_node_update>
const int INF = 1e10;
const int p=39;
const int mod = 2286661337;
const int maxn = 300000;
mt19937_64 gen(time(nullptr));
vector<int> ans(maxn);
int t = 0;
void dfs(int v, vector<deque<int>>& g) {
    if (t > 100000) {
        return;
    }
    int to = g[v][0];
    g[v].pop_front();
    g[v].push_back(to);
    ans[t] = v;
    t++;
    dfs(to, g);
}
signed main() {
    ios_base::sync_with_stdio(false);
    cin.tie(nullptr);
    int n, q;
    cin >> n >> q;
    vector<deque<int>> g(n);
    for (int i = 0; i < n; i++) {
        int c;
        cin >> c;
        for (int j = 0; j < c; j++) {
            int to;
            cin >> to;
            to--;
            g[i].push_back(to);
        }
    }
    vector<int> zap(q);
    for (int i = 0; i < q; i++) {
        cin >> zap[i];
    }
    int v = 0;
    for (int t = 0; t <= 100000; t++) {
        int to = g[v][0];
```

```
        g[v].pop_front();
        g[v].push_back(to);
        ans[t] = v;
        v = to;
    }
    for (int i = 0; i < q; i++) {
        cout << ans[zap[i]]+1 << " ";
    }
}
```

Task F

```
#include <bits/stdc++.h>
#include <ext/pb_ds/assoc_container.hpp>
using namespace std;
using namespace __gnu_pbds;
#define ordered_set tree<int, null_type, less<int>, rb_tree_tag,
    tree_order_statistics_node_update>
const int INF = 1e10;
const int p=39;
const int mod = 2286661337;
const int maxn = 300000;
mt19937_64 gen(time(nullptr));
#define int long long
signed main() {
    ///
    ios_base::sync_with_stdio(false);
    cin.tie(nullptr);
    int h, w;
    cin >> h >> w;
    vector<vector<int>> pole(h, vector<int>(w));
    for (int i = 0; i < h; i++) {
        string s;
        cin >> s;
        for (int j = 0; j < w; j++) {
            pole[i][j] = s[j]-'0';
        }
    }
    int ans1 = 0, ans2 = 0, ans3 = 0, ans4 = 0, ans5 = 0, ans6 = 0;
    for (int i = 0; i < h; i+=2) {
        for (int j = 0; j < w; j++) {
            ans1 += pole[i][j]*pole[i+1][j];
        }
    }
    for (int i = 0; i < h; i++) {
        for (int j = 0; j < w; j+=2) {
            ans2 += pole[i][j]*pole[i][j+1];
        }
    }
    for (int i = 0; i < h; i+=2) {
        for (int j = 0; j+3 <= w-1; j+=4) {
            ans3 += pole[i][j]*pole[i+1][j];
            ans3 += pole[i][j+1]*pole[i][j+2];
```

```
        ans3 += pole[i+1][j+1]*pole[i+1][j+2];
        ans3 += pole[i][j+3]*pole[i+1][j+3];
    }
    if (w%4!=0) {
        ans3 += pole[i][w-1]*pole[i+1][w-1];
        ans3 += pole[i][w-2]*pole[i+1][w-2];
    }
}
for (int i = 0; i < h; i++) {
    for (int j = 1; j < w-1; j+=2) {
        ans4 += pole[i][j]*pole[i][j+1];
    }
    ans4 += pole[i][0]*pole[i][w-1];
}
for (int i = 1; i < h-1; i++) {
    for (int j = 1; j < w-1; j+=2) {
        ans5 += pole[i][j]*pole[i][j+1];
    }
    ans5 += pole[i][0]*pole[i][w-1];
}
for (int i = 0; i < w; i++) {
    ans5 += pole[0][i]*pole[h-1][i];
}
vector<vector<char>> pep(h, vector<char>(w));
for (int i = 0; i < h; i+= 2) {
    for (int j = 0; j < w; j+=2) {
        int fst = pole[i][j]*pole[i][j+1];
        fst += pole[i+1][j]*pole[i+1][j+1];
        int scnd = pole[i+1][j]*pole[i][j];
        scnd += pole[i+1][j+1]*pole[i][j+1];
        if (scnd > fst) {
            ans6 += scnd;
            pep[i][j] = 'T';
            pep[i][j+1] = 'T';
            pep[i+1][j] = 'B';
            pep[i+1][j+1] = 'B';
        } else {
            ans6 += fst;
            pep[i][j] = 'L';
            pep[i][j+1] = 'R';
            pep[i+1][j] = 'L';
            pep[i+1][j+1] = 'R';
        }
    }
}
vector<int> mx;
mx.push_back(ans1);
mx.push_back(ans2);
mx.push_back(ans3);
mx.push_back(ans4);
mx.push_back(ans5);
mx.push_back(ans6);
sort(mx.begin(), mx.end());
```

```
if (mx.back()==ans1) {
    cout << ans1 << "\n";
    for (int i = 0; i < h; i+=2) {
        for (int j = 0; j < w; j++) {
            cout << "T";
        }
        cout << "\n";
        for (int j = 0; j < w; j++) {
            cout << "B";
        }
        cout << "\n";
    }
} else if (mx.back()==ans2){
    cout << ans2 << "\n";
    for (int i = 0; i < h; i++) {
        for (int j = 0; j < w; j+=2) {
            cout << "LR";
        }
        cout << "\n";
    }
} else if (mx.back()==ans3){
    cout << ans3 << "\n";
    for (int i = 0; i < h; i+=2) {
        for (int j = 0; j+3 <= w-1; j+=4) {
            cout << "TLRT";
        }
        if (w%4!=0) {
            cout << "TT";
        }
        cout << "\n";
        for (int j = 0; j+3 <= w-1; j+=4) {
            cout << "BLRB";
        }
        if (w%4!=0) {
            cout << "BB";
        }
        cout << "\n";
    }
} else if (mx.back()==ans4) {
    cout << ans4 << "\n";
    for (int i = 0; i < h; i++) {
        cout << "R";
        for (int j = 1; j < w-1; j+=2) {
            cout << "LR";
        }
        cout << "L";
        cout << "\n";
    }
} else if (mx.back()==ans5) {
    cout << ans5 << "\n";
    for (int i = 0; i < w; i++) {
        cout << "B";
    }
```

```
        cout << "\n";
        for (int i = 1; i < h-1; i++) {
            cout << "R";
            for (int j = 1; j < w-1; j+=2) {
                cout << "LR";
            }
            cout << "L";
            cout << "\n";
        }
        for (int i = 0; i < w; i++) {
            cout << "T";
        }
    } else if (mx.back() ==ans6) {
        cout << ans6 << "\n";
        for (int i = 0; i < h; i++) {
            for (int j = 0; j < w; j++) {
                cout << pep[i][j];
            }
            cout << "\n";
        }
    }
}
```