

**Решения заданий заключительного этапа
открытой олимпиады школьников по программированию
«Когнитивные технологии»**

14.01.2018 г.

За каждую правильно решенную задачу участник получал "+".

В таблице результатов все участники ранжировались сначала по количеству решенных задач от большего к меньшему, внутри каждой задачи по затраченному времени.

Задача А. Шифровка клавиатурой

```
#include <iostream>
#include <string>
#include <vector>

using namespace std;

vector <string> vec = {
    "qwertyuiop",
    "asdfghjkl",
    "zxcvbnm"
};

int main() {
    string s;
    int k;
    cin >> s >> k;
    for (int i = 0; i < s.size(); ++i) {
        for (int j = 0; j < vec.size(); ++j) {
            int t;
            for (t = 0; t < vec[j].size(); ++t) {
                if (vec[j][t] == s[i]) {
                    break;
                }
            }
            if (t < vec[j].size()) {
                cout << vec[j][(t + k) % vec[j].size()];
            }
        }
    }
    return 0;
}
```

Задача В. Хитрая функция

```
#include <iostream>
#include <stdio.h>
#include <cstring>
#include <vector>
```

```

#include <algorithm>
#include <cstdlib>
#include <cmath>
#include <queue>
#include <stack>
#include <map>
#include <set>
#include <ctime>
#include <cassert>
#include <unordered_map>
#include <fstream>
#include <random>
#include <cstring>
#include <bitset>
#include <functional>

#define all(a) (a).begin(), (a).end()
#define pb push_back
#define sz(a) (int)(a).size()

using namespace std;

typedef long long ll;
typedef pair<int, int> pii;
typedef pair<ll, ll> pll;
typedef long double ld;

int p;

void add(int &a, int b) {
    a += b;
    if (a >= p) {
        a -= p;
    }
}

void mul(int &a, int b) {
    ll c = (ll)a * b;
    c %= p;
    a = c;
}

int binPow(int a, int b) {
    if (b == 0) {
        return 1;
    }
    int res = binPow(a, b / 2);
    mul(res, res);
    if (b & 1) {
        mul(res, a);
    }
    return res;
}

int f(int a, int b, int x, int y) {
    int tmp, u = 0, v = 0;

    tmp = a;
    mul(tmp, x);
    add(u, tmp);
    tmp = b;

```

```

    mul(tmp, y);
    add(u, tmp);

    tmp = a;
    mul(tmp, b);
    add(v, tmp);
    tmp = x;
    mul(tmp, y);
    add(v, tmp);

    if (v == 0) {
        return 1;
    }
    int res = u;
    mul(res, binPow(v, p - 2));
    return res;
}

int main() {
    // ifstream cin("input.txt");
    // ofstream cout("output.txt");
    // freopen("input.txt", "r", stdin);
    // freopen("output.txt", "w", stdout);
    ios_base::sync_with_stdio(false);
    cin.tie(0);
    cout.tie(0);

    int a, b;
    ll k;
    cin >> a >> b >> p >> k;

    int cur = 0;

    if (a == 0) {
        cur = k % p;
        k = min(k, 2 + k % 2);
    } else {
        k = min((ll)a + 1, k);
        cur = k;
    }

    for (int i = k - 1; i >= 1; --i) {
        cur = f(a, b, cur, i);
    }

    cout << cur << endl;
    return 0;
}

```

Задача С. Список фильмов

```

#include <iostream>
#include <vector>
#include <cstdio>

using namespace std;

vector <int> a, b;

```

```

int main() {
    ios_base::sync_with_stdio(0);
    cin.tie(0);
    int n;
    cin >> n;

    a.resize(n);
    b.resize(n);

    for (int i = 0; i < n; ++i) {
        cin >> a[i] >> b[i];
    }

    bool fl = 1;
    for (int i = 0; i + 1 < n; ++i) {
        if (a[i] > a[i + 1] && b[i] > b[i + 1]) {
            fl = 0;
            break;
        }
    }

    if (!fl) {
        cout << "NO" << endl;
        return 0;
    }
    cout << "YES" << endl;
    for (int i = n - 1; i >= 0; --i) {
        cout << i + 1 << " ";
        if (i == n - 1 || a[i] <= a[i + 1]) {
            cout << "1\n";
        } else {
            cout << "2\n";
        }
    }
    return 0;
}

```

Задача D. Плюсы и минусы последовательностей

```

#include <iostream>
#include <stdio.h>
#include <cstring>
#include <vector>
#include <algorithm>
#include <cstdlib>
#include <cmath>
#include <queue>
#include <stack>
#include <map>
#include <set>
#include <ctime>
#include <cassert>
#include <unordered_map>
#include <fstream>
#include <random>
#include <cstring>
#include <bitset>
#include <functional>

```

```

#define all(a) (a).begin(), (a).end()
#define sz(a) (int)(a).size()

using namespace std;

typedef long long ll;
typedef pair<int, int> pii;
typedef pair<ll, ll> pll;
typedef long double ld;

const int P = 1e9 + 7;

int main() {
    // ifstream cin("input.txt");
    // ofstream cout("output.txt");
    // freopen("input.txt", "r", stdin);
    // freopen("output.txt", "w", stdout);
    ios_base::sync_with_stdio(false);
    cin.tie(0);
    cout.tie(0);

    string s;
    cin >> s;
    int n = sz(s);

    vector<int> p(n);
    for (int i = 0; i < n; ++i) {
        p[i] = i;
    }

    int res = 0;
    for ( ; ; ) {

        bool ok = true;

        for (int i = 0; i < n; ++i) {
            if (p[i] >= i && s[i] == '-') {
                ok = false;
                break;
            }
            if (p[i] <= i && s[i] == '+') {
                ok = false;
                break;
            }
        }
        res += ok;

        if (!next_permutation(all(p))) {
            break;
        }
    }

    cout << res << endl;
    return 0;
}

```

Задача Е. Подземелья, подземелья и ещё больше подземелий

```
#include <cstdio>
#include <iostream>
#include <ctime>
#include <string>
#include <vector>
#include <cmath>
#include <algorithm>
#include <cstring>
#include <set>
#include <map>
#include <cstdlib>
#include <ctime>
#include <cassert>
#include <bitset>
#include <deque>
#include <stack>
#include <climits>
#include <string>
#include <queue>
#include <memory.h>

#define ll long long
#define ld double
#define pii pair<int, int>
#define forn(i, n) for (int i = 0; i < (int)n; i++)
#define mp make_pair
#define ui unsigned ll
#define pll pair<ll, ll>

using namespace std;

const int maxn = (int)1e5 + 10;
int v[maxn];

const int inf = (int)1e9 + 10;

int calc(int n, int x) {
    ll ans = 0;

    for (int i = n; i >= 0; i--) {
        ans *= x;
        ans += v[i];
        ans = min(ans, (ll)inf);
    }

    return (int)ans;
}

int main() {
    int n;

    int t;

    cin >> n >> t;

    for (int i = 0; i <= n; i++) {
        scanf("%d", &v[i]);
    }
}
```

```

int l = -1;
int r = 1e9 + 10;

while (l != r) {
    ll m = (l + r + 1) >> 1;

    if (calc(n, m) <= t) {
        l = m;
    }
    else {
        r = m - 1;
    }
}

int q;

cin >> q;

while (q--) {
    int x;

    scanf("%d", &x);

    if (x <= l) {
        printf("YES\n");
    }
    else {
        printf("NO\n");
    }
}

return 0;
}

```

Задача F. Подарки

```

#include <cstdio>
#include <iostream>
#include <ctime>
#include <string>
#include <vector>
#include <cmath>
#include <algorithm>
#include <cstring>
#include <set>
#include <map>
#include <cstdlib>
#include <ctime>
#include <cassert>
#include <bitset>
#include <deque>
#include <stack>
#include <climits>
#include <string>
#include <queue>
#include <memory.h>

#define ll long long

```

```

#define ld double
#define pii pair <int, int>
#define forn(i, n) for (int i = 0; i < (int)n; i++)
#define mp make_pair
#define ui unsigned ll
#define pll pair <ll, ll>

using namespace std;

const int maxn = (int)6e4 + 10;

char s[maxn];
bitset <maxn> b;
int cnt[maxn];
bitset <maxn> now;

int main() {
    int n;

    cin >> n;

    for (int i = 0; i < 2 * n + 1; i++) {
        char x;

        scanf(" %c", &x);

        now[i] = x - '0';
    }

    scanf("%s", s);

    int pos = n;

    for (int i = 0; i < n; i++) {
        if (s[i] == 'T') {
            cnt[pos]++;
            b[pos] = 1;
        }
        else if (s[i] == 'R') {
            pos++;
        }
        else {
            pos--;
        }
    }

    int dx = pos - n;

    int ans = (now & b).count();

    int st = n;

    for (int i = 0; i < n - 1; i++) {
        if (s[i] == 'L') {
            b <<= 1;
            st--;
        }
        else if (s[i] == 'R') {
            b >>= 1;
            st++;
        }
    }
}

```

```

        else {
            if (cnt[st] == 0) {
                throw 1;
            }

            cnt[st]--;
            if (cnt[st] == 0) {
                b[n] = 0;
            }

            cnt[st + dx]++;

            if (cnt[st + dx] == 1) {
                b[n + dx] = 1;
            }
        }

        ans = max(ans, (int)(b & now).count());
    }

    cout << ans << endl;

    return 0;
}

```

Задача G. Бегущие по стадиону

```

#include <iostream>
#include <stdio.h>
#include <cstring>
#include <vector>
#include <algorithm>
#include <cstdlib>
#include <cmath>
#include <queue>
#include <stack>
#include <map>
#include <set>
#include <ctime>
#include <cassert>
#include <unordered_map>
#include <fstream>
#include <random>
#include <cstring>
#include <complex>
#include <bitset>

#define all(a) (a).begin(), (a).end()
#define pb push_back
#define sz(a) (int)(a).size()

using namespace std;

typedef long long ll;
typedef pair<int, int> pii;
typedef pair<ll, ll> pll;
typedef long double ld;

int main() {

```

```

// ifstream cin("input.txt");
// ofstream cout("output.txt");
// freopen("input.txt", "r", stdin);
// freopen("output.txt", "w", stdout);

ios_base::sync_with_stdio(false);
cin.tie(0);
cout.tie(0);

int n, len;
cin >> n >> len;

vector<int> x(n), v(n);

for (int i = 0; i < n; ++i) {
    cin >> x[i] >> v[i];
}

vector<int> p(n);
for (int i = 0; i < n; ++i) {
    cin >> p[i];
    --p[i];
}

vector<pair<ll, int>> event;
int cur = 0;

for (int k = 0; k + 1 < n; ++k) {
    int i = p[k];
    int j = p[k + 1];

    ll tm = 0;

    if (v[i] == -1 && v[j] == -1) {
        tm = x[i] + x[j];
    } else if (v[i] == 1 && v[j] == 1) {
        tm = (len - x[i]) + (len - x[j]);
    } else if (v[i] == 1 && v[j] == -1) {
        tm = abs(x[i] - x[j]);
        tm = (x[i] < x[j] ? tm : 2 * len - tm);
    } else {
        tm = abs(x[i] - x[j]);
        tm = (x[j] < x[i] ? tm : 2 * len - tm);
    }

    if (x[i] < x[j] || (x[i] == x[j] && v[i] < v[j])) {
        ++cur;
        event.push_back({tm, -1});
        event.push_back({tm + 2 * len, 1});
    } else {
        event.push_back({tm, 1});
        event.push_back({tm + 2 * len, -1});
    }
}

event.push_back({0, 0});
sort(all(event));

for (int i = 0; i < sz(event); ) {
    int j = i;
    for (; j < sz(event) && event[i].first == event[j].first; ++j) {

```

```

        cur += event[j].second;
    }
    // cout << event[i].first << " " << cur << endl;

    if (cur == n - 1 && event[i].first != 4 * len) {
        cout << "Yes" << endl;
        return 0;
    }
    i = j;
}
cout << "No" << endl;
return 0;
}

```

Задача Н. Свитера Мейбл

```

#include <cstdio>
#include <iostream>
#include <ctime>
#include <string>
#include <vector>
#include <cmath>
#include <algorithm>
#include <cstring>
#include <set>
#include <map>
#include <cstdlib>
#include <ctime>
#include <cassert>
#include <bitset>
#include <deque>
#include <stack>
#include <climits>
#include <string>
#include <queue>
#include <memory.h>

#define ll long long
#define ld double
#define pii pair<int, int>
#define forn(i, n) for (int i = 0; i < (int)n; i++)
#define mp make_pair
#define ui unsigned ll
#define pll pair<ll, ll>

using namespace std;

vector<pair<pii, int>> v;

char s[10];

int cnt[3];
int f[3];

int main() {
    for (int i = 0; i < 3; i++) {
        int x, y;

        scanf("%d %d", &x, &y);
    }
}

```

```

        v.push_back(mp(mp(x, y), i));
    }

    sort(v.rbegin(), v.rend());

    for (int i = 0; i < 3; i++) {
        f[v[i].second] = i;
    }

    int n;

    cin >> n;

    for (int i = 0; i < n; i++) {
        scanf("%s", s);

        vector <int> g;

        if (s[0] == 'R' || s[1] == 'R') {
            g.push_back(0);
        }

        if (s[0] == 'G' || s[1] == 'G') {
            g.push_back(1);
        }

        if (s[0] == 'B' || s[1] == 'B') {
            g.push_back(2);
        }

        g[0] = f[g[0]];
        g[1] = f[g[1]];

        sort(g.begin(), g.end());

        if (g[0] == 0) {
            if (g[1] == 1) {
                cnt[0]++;
            }
            else {
                cnt[1]++;
            }
        }
        else {
            cnt[2]++;
        }
    }

    ll ans = -1;

    for (int i = 0; i <= cnt[0]; i++) {
        if (i > v[0].first.second) {
            continue;
        }

        if (cnt[0] - i > v[1].first.second) {
            continue;
        }

        ll now = (ll)v[0].first.first * i;
    }

```

```

        now += (ll)v[1].first.first * (cnt[0] - i);

        int a = v[0].first.second - i;
        int b = v[1].first.second - (cnt[0] - i);
        int c = v[2].first.second;

        int cn1 = cnt[1];
        int cn2 = cnt[2];

        if (cn1 <= a) {
            now += (ll)v[0].first.first * cn1;
        }
        else {
            now += (ll)v[0].first.first * a;
            cn1 -= a;

            if (c < cn1) {
                continue;
            }

            now += (ll)v[2].first.first * cn1;
            c -= cn1;
        }

        if (cn2 <= b) {
            now += (ll)v[1].first.first * cn2;
        }
        else {
            now += (ll)v[1].first.first * b;
            cn2 -= b;

            if (c < cn2) {
                continue;
            }

            now += (ll)v[2].first.first * cn2;
        }

        ans = max(ans, now);
    }

    if (ans == -1) {
        cout << "NO" << endl;
    }
    else {
        cout << "YES" << endl;
        cout << ans << endl;
    }

    return 0;
}

```

Задача I. Диппер и таблица

```

#include <iostream>
#include <set>
#include <map>
#include <vector>
#include <string>

```

```

#include <algorithm>
#include <cmath>
#include <cstdio>
#include <climits>
#include <ctime>
#include <cassert>
#include <queue>
#include <deque>
#include <stack>
#include <memory.h>
#include <bitset>
#include <cstring>
#include <assert.h>

using namespace std;

#define ll long long
#define mp make_pair

const double pi = 3.14159265358979323846;
const double eps = 1e-6;
const ll maxn = 100010;
const ll inf = 1000000007;
const ll mod = 1000000007;
#define ld double

int n, m, x, c, a[maxn], b[maxn];
vector<vector<int>> > v(maxn);

ll bpow(ll x, ll n) {
    if (n == 0) {
        return 1;
    }
    else if (n == 1) {
        return x;
    }
    else if (n % 2 == 0) {
        ll tmp = bpow(x, n / 2);
        return (tmp * tmp) % mod;
    }
    else {
        return (x * bpow(x, n - 1)) % mod;
    }
}

int main() {
    cin >> n >> m >> c;
    for (int i = 0; i < n; i++) {
        for (int j = 0; j < m; j++) {
            cin >> x;
            v[i].push_back(x);
            a[i] = max(a[i], x);
            b[j] = max(b[j], x);
        }
    }
    int mx = 0;
    for (int i = 0; i < n; i++) {
        for (int j = 0; j < m; j++) {
            if (v[i][j] != min(a[i], b[j])) {
                cout << 0;
            }
        }
    }
}

```

```

        return 0;
    }
    mx = max(mx, v[i][j]);
}
}
int k1 = 0, k2 = 0;
for (int i = 0; i < n; i++) {
    if (a[i] == mx) {
        k1++;
    }
}
for (int i = 0; i < m; i++) {
    if (b[i] == mx) {
        k2++;
    }
}
ll ans = (bpow(c - mx + 1, k1) + bpow(c - mx + 1, k2) - 1 + mod) %
mod;
cout << ans;

return 0;
}

```

Задача J. Региональный вопрос

```

#include <bits/stdc++.h>

using namespace std;

#define left soidvjsdivj
#define right sdosjdovidfjivf
#define less osijvdiv
#define equal dkpcmdoiv

#define sz(x) ((int)(x).size())

const int maxn = 1e5 + 10;
const int mod = 1e9 + 7;

namespace dsu {
    int parent[maxn];
    int rk[maxn];
    int root[maxn];

    void init() {
        for (int i = 0; i < maxn; ++i) {
            parent[i] = i;
            root[i] = i;
            rk[i] = 1;
        }
    }

    int find_set(int v) {
        if (v == parent[v]) return v;
        return parent[v] = find_set(parent[v]);
    }

    void unite(int a, int b) {
        a = find_set(a);
    }
}

```

```

        b = find_set(b);
        if (a != b) {
            int rt = root[a];
            if (rk[a] < rk[b]) swap(a, b);
            parent[b] = a;
            rk[a] += rk[b];
            root[a] = rt;
        }
    }
};

int n, k;
vector<int> graph[maxn];
char marked[maxn];
int height[maxn];
vector<int> layer[maxn];
char is_marked_layer[maxn];
int last_marked[maxn];
int maxh;
int size[maxn];
int left[maxn], right[maxn];
int dp[maxn];
int suf[maxn];

void dfs1(int v, int p) {
    if (p != -1) height[v] = height[p] + 1;
    layer[height[v]].push_back(v);
    if (marked[v]) {
        is_marked_layer[height[v]] = true;
    }
    for (auto to: graph[v]) {
        if (to != p) {
            dfs1(to, v);
        }
    }
}

void precalc() {
    dfs1(1, -1);
    maxh = *max_element(height, height + maxn);
    for (int i = 0; i <= maxh; ++i) {
        if (is_marked_layer[i]) {
            last_marked[i] = i;
        } else {
            last_marked[i] = (i == 0) ? -1: last_marked[i - 1];
        }
    }
}

int fixedval;
int less, equal, more;

void clear_all() {
    dsu::init();
    memset(size, 0, sizeof(size));
    memset(left, 0, sizeof(left));
    memset(right, 0, sizeof(right));
    less = equal = more = 0;
}

void update_left(int h) {

```

```

    less = equal = more = 0;
    for (auto v: layer[h]) {
        size[v] = marked[v];
        for (auto to: graph[v]) {
            if (height[to] < height[v]) continue;
            dsu::unite(v, to);
            size[v] += size[to];
        }
        if (size[v] > fixedval) ++more;
        else if (size[v] == fixedval) ++equal;
        else ++less;
    }
}

void update_right(int h) {
    for (auto v: layer[h]) {
        int rt = dsu::root[dsu::find_set(v)];
        if (!marked[v]) continue;
        if (size[rt] > fixedval) --more;
        else if (size[rt] == fixedval) --equal;
        else --less;
        --size[rt];
        if (size[rt] > fixedval) ++more;
        else if (size[rt] == fixedval) ++equal;
        else ++less;
    }
}

int solve(int need) {
    clear_all();
    fixedval = need;
    for (int i = maxh, p = maxh; i >= 0; --i) {
        update_left(i);
        while (more > 0) update_right(p--);
        if (equal == sz(layer[i])) {
            right[i] = p + 1;
            left[i] = max(i, last_marked[p]) + 1;
        }
    }
    dp[maxh + 1] = 1;
    suf[maxh + 1] = 1;
    for (int i = maxh; i >= 0; --i) {
        if (right[i]) {
            dp[i] = suf[left[i]] - suf[right[i] + 1];
            if (dp[i] < 0) dp[i] += mod;
        } else {
            dp[i] = 0;
        }
        suf[i] = dp[i] + suf[i + 1];
        if (suf[i] >= mod) suf[i] -= mod;
    }
    return dp[0];
}

signed main() {
    ios_base::sync_with_stdio(false); cin.tie(0);

    cin >> n >> k;
    for (int i = 0; i < k; ++i) {
        int v; cin >> v;
        marked[v] = true;
    }
}

```

```

    }
    for (int i = 1; i < n; ++i) {
        int a, b;
        cin >> a >> b;
        graph[a].push_back(b);
        graph[b].push_back(a);
    }

    precalc();
    int ans = 0;
    for (int d = 1; d <= k; ++d) {
        if (k % d == 0) {
            ans += solve(d);
            if (ans >= mod) ans -= mod;
        }
    }
    cout << ans << '\n';
}

```