

Олимпиада СПбГУ по информатике 2018/19 учебного года

Илаев Никита Михайлович

A	B	C	D	E	F	Sum
100	100	100	60	0	0	360

Task A (100)

```
#include <bits/stdc++.h>

using namespace std;

typedef long long ll;

#define fi first
#define se second
#define V vector
#define pb push_back
#define forn(i, n) for (int (i) = 0; (i) < (n); (i)++)

int main() {
    ll n, m;
    cin >> n >> m;
    bool fl = false;
    if (n == m) fl = true;
    else if (m > n && m % n == 0){
        m /= n;
        while (m % 2 == 0) m /= 2;
        if (m == 1) fl = true;
    }

    if (fl) cout << "Yes";
    else cout << "No";
    return 0; }
```

Task B (100)

```
#include <bits/stdc++.h>

using namespace std;

typedef long long ll;

#define fi first
#define se second
#define V vector
#define pb push_back
#define forn(i, n) for (int (i) = 0; (i) < (n); (i)++)

int main() {
    int n;
    cin >> n;
    string s;
    cin >> s;
    bool ans = false;
    forn(i, n){
        if (s[i] == 'r'){
            if (i > 0 && s[i-1] == 'o'){
                ans = true;
                break;}
            if (i > 1 && s[i-2] == 'o'){
                ans = true;
                break;}
            if (i < n-1 && s[i+1] == 'o'){
                ans = true;
                break;}
        }
    }
    if (ans) cout << "Yes";
    else cout << "No";
    return 0; }
```

Task C (100)

```
#include <bits/stdc++.h>

using namespace std;

#define ll long long
#define ull unsigned long long
#define ld long double
#define fr(i, n) for(int i = 0; i < n; ++i)
#define fr_o(i, n) for(int i = 1; i < n; ++i)
#define inp(a, n) for(int i = 0; i < n; cin >> a[i++]);
#define ret return 0;
#define skip cout << '\n';
#define fast ios_base::sync_with_stdio(0); cin.tie(0); cout.tie(0);

void calc(ll v, vector <vector <ll>> &nodes, vector <ll> &dp, ll p) {
    dp[v] = 1;
    fr(i, nodes[v].size()) {
        if(nodes[v][i] != p) {
            calc(nodes[v][i], nodes, dp, v);
            dp[v] += dp[nodes[v][i]];
        }
    }
}

ll ans[100005];

void solve(ll v, vector <vector <ll>> &nodes, vector <ll> &dp, ll p, ll up) {
    ll maxim = up;
    up += dp[v];
    fr(i, nodes[v].size()) {
        if(nodes[v][i] != p) {
            maxim = max(maxim, dp[nodes[v][i]]);
        }
    }
    ans[v] = maxim + 1;
    fr(i, nodes[v].size()) {
        if(nodes[v][i] != p) solve(nodes[v][i], nodes, dp, v, up - dp[nodes[v][i]]);
    }
}

int main() {
    fast
    ll n;
    cin >> n;
    vector <vector <ll>> nodes (n);
    fr(i, n - 1) {
        ll a, b;
        cin >> a >> b;
        nodes[a - 1].push_back(b - 1);
        nodes[b - 1].push_back(a - 1);
    }
    vector <ll> dp (n);
    calc(0, nodes, dp, 0);
    solve(0, nodes, dp, 0, 0);
    fr(i, n) cout << ans[i] << ' ';
}
```

Task D (60)

```
#include <bits/stdc++.h>

using namespace std;

#define ll long long
#define ull unsigned long long
#define ld long double
#define fr(i, n) for(int i = 0; i < n; ++i)
#define fr_o(i, n) for(int i = 1; i < n; ++i)
#define inp(a, n) for(int i = 0; i < n; cin >> a[i++]);
#define ret return 0;
#define skip cout << '\n';
#define fast ios_base::sync_with_stdio(0); cin.tie(0); cout.tie(0);

int main() {
    fast
    string q;
    cin >> q;
    ll t, n, p;
    cin >> t >> n >> p;
    if(q == "split") {
        fr(1, t) {
            string s;
            cin >> s;
            fr(i, 6) cout << s[i];
            cout << "a_";
            for(int i = 3; i < s.size(); ++i) cout << s[i];
            cout << "b_";
            cout << s[0] << s[1] << s[2] << s[6] << s[7] << s[8];
            cout << "c_";
            if(n == 5) {
                cout << s[0] << s[1] << s[2] << s[6] << s[7] << s[8];
                cout << "d_";
                cout << s[1] << s[2] << s[3] << s[4] << s[5] << s[6];
                cout << "e";
            }
            cout << '\n';
        }
    }
    else {
        fr(1, t) {
            vector<string> v (n / 2 + 1);
            bool a = false, b = false;
            fr(i, v.size()) {
                cin >> v[i];
                a |= (v[i][v[i].size() - 1] == 'a');
                b |= (v[i][v[i].size() - 1] == 'b');
            }
            if(a) {
                fr(i, v.size()) {
                    if(v[i][v[i].size() - 1] == 'a') {
                        cout << v[i][0] << v[i][1] << v[i][2] << v[i][3] << v[i][4] << v[i][5];
                    }
                }
                fr(i, v.size()) {
                    if(v[i][v[i].size() - 1] != 'a' and v[i][v[i].size() - 1] != 'e') {
                        cout << v[i][3] << v[i][4] << v[i][5];
                        break;
                    }
                }
            }
            else if(b) {
                fr(i, v.size()) {
                    if(v[i][v[i].size() - 1] != 'b' and v[i][v[i].size() - 1] != 'e') {
                        cout << v[i][0] << v[i][1] << v[i][2];
                        break;
                    }
                }
                fr(i, v.size()) {

```

```

        if(v[i][v[i].size() - 1] == 'b') {
            cout << v[i][0] << v[i][1] << v[i][2] << v[i][3] << v[i][4] << v[i][5];
        }
    }
    else {
        fr(i, v.size()) {
            if(v[i][v[i].size() - 1] != 'e') {
                cout << v[i][0];
                break;
            }
        }
        fr(i, v.size()) {
            if(v[i][v[i].size() - 1] == 'e') {
                cout << v[i][0] << v[i][1] << v[i][2] << v[i][3] << v[i][4] << v[i][5];
                break;
            }
        }
        fr(i, v.size()) {
            if(v[i][v[i].size() - 1] != 'e') {
                cout << v[i][4] << v[i][5];
                break;
            }
        }
    }
    cout << '\n';
}
}
}

```

Task E (—)

Task F (—)