Олимпиада СПбГУ по информатике 2018/19 учебного года

Анчутин Никита Сергеевич

A	B	C	D	E	F	Sum
100	100	100	30	35	7	372

Task A (100)

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

using namespace std;

#define x first
#define y second
typedef long long ll;

int main()
{
    ios::sync_with_stdio(false);
    cin.tie(NULL);

    ll n, m;
    cin >> n >> m;

    while (m > n) {
        n *= 2;
    }

    if (n == m) {
        cout << "Yes";
    } else {
        cout << "No";
    }
    return 0;
}
```

Task B (100)

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

using namespace std;

#define x first
#define y second
typedef long long ll;

int main()
{
    ios::sync_with_stdio(false);
    cin.tie(NULL);

    ll n;
    cin >> n;

    string s;
    cin >> s;

    for (int i = 0; i < n - 1 ; i++) {
        if (s[i] == 'o' and s[i + 1] == 'r' or s[i] == 'r' and s[i + 1] == 'o') {
            return cout << "Yes", 0;
        }

        if (i < n - 2) {
            if (s[i] == 'o' and s[i + 2] == 'r') {
                return cout << "Yes", 0;
            }
        }
    }

    cout << "No";

    return 0;
}
```

Task C (100)

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

using namespace std;

#define x first
#define y second
typedef long long ll;

vector<vector<ll> > graph;
vector<bool> used;
vector<ll> ans, s;
ll n, olala;

void dfs (ll v) {
    used[v] = true;

    bool check = false;

    for (int i = 0; i < graph[v].size(); i++) {
        if (!used[graph[v][i]]) {
            check = true;
            dfs(graph[v][i]);
        }
    }

    //cout << v << " " << check << '\n';

    s[v] = 1;
    ans[v] = 1;

        ll mx = 0;

    for (int i = 0; i < graph[v].size(); i++) {
        mx = max(mx, s[graph[v][i]]);
        s[v] += s[graph[v][i]];
        //ans[v] += ans[graph[v][i]];
    }
    ans[v] += max(mx, n - s[v]);

}


int main()
{
    ios::sync_with_stdio(false);
    cin.tie(NULL);

    ll a, b;
    cin >> n;

    graph.resize(n);
    used.resize(n);
    ans.resize(n);
    s.resize(n);
    if (n == 1) {
        return cout << 1, 0;
    }

    for (int i = 0; i < n - 1; i++) {
        cin >> a >> b;
        a--;
        b--;
        graph[a].push_back(b);
        graph[b].push_back(a);
    }


    for (int i = 0; i < n; i++) {
        if (graph[i].size() == 1) {
```

```
            //cout << i << "\n";
            dfs(i);
            break;
        }
    }

    for (int i = 0; i < n; i++) {
        //cout << ans[i] << " " << s[i] << '\n';
        //if (graph[i].size() > 2 and ans[i] == 2) {
        //    cout << 2 << " ";
        //} else {
            cout << ans[i] << "␣";
        //}
    }


    return 0;
}
```

Task D (30)

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

using namespace std;

#define x first
#define y second
typedef long long ll;


int main()
{
    ios::sync_with_stdio(false);
    cin.tie(NULL);

    //freopen("input.txt", "r");

    string s;
    cin >> s;
    if (s == "split") {
        ll t, n, p;
        cin >> t >> n >> p;

        while (t--) {
            cin >> s;
            cout << s.substr(0, 6) << "s" <<  "␣" << s.substr(0, 3) << s.substr(6, 9) << "m" << "␣
                " <<  s.substr(3, 9) << "f" << '\n';
        }
    } else {
        ll t, n, p;
        cin >> t >> n >> p;
        string s1, s2;
        while (t--) {
            cin >> s1 >> s2;

            if (s1[6] == 's') {
                cout << s1.substr(0, 6) << s2[3] << s2[4] << s2[5];
            } else if (s2[6] == 's') {
                cout << s2.substr(0, 6) << s1[3] << s1[4] << s1[5];
            } else if (s1[6] == 'm') {
                cout << s1[0] << s1[1] << s1[2] << s2.substr(0, 6);
            } else {
                cout << s2[0] << s2[1] << s2[2] << s1.substr(0, 6);
            }

            cout << '\n';

        }
    }


    return 0;
}
```

Task E (35)

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

using namespace std;

#define x first
#define y second
typedef long long ll;


int main()
{
    ios::sync_with_stdio(false);
    cin.tie(NULL);

    ll n, a, b, c, d, mx = -1e10, mn = 1e10;
    cin >> n;
    pair<ll, ll> s, f;
    vector<pair<ll, ll> > spisok1, spisok2;

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

        if (a > mx) {
            mx = a;
            spisok1.clear();
            spisok1.push_back({abs(b), i + 1});
        } else if (a == mx) {
            spisok1.push_back({abs(b), i + 1});
        }

        if (a < mn) {
            mn = a;
            spisok2.clear();
            spisok2.push_back({abs(b), i + 1});
        } else if (a == mn) {
            spisok2.push_back({abs(b), i + 1});
        }

        //spisok.push_back({{a, abs(b)}, i + 1});
    }

    cin >> a >> b >> c >> d;



    if (a < c) {
        sort(spisok1.begin(), spisok1.end());

        if (spisok1.size() == 1) {
            cout << spisok1[0].y;
        } else if (spisok1[0].x == spisok1[1].x) {
            cout << -1;
        } else {
            cout << spisok1[0].y;
        }

    } else {
        sort(spisok2.begin(), spisok2.end());

        if (spisok2.size() == 1) {
            cout << spisok2[0].y;
        } else if (spisok2[0].x == spisok2[1].x) {
            cout << -1;
        } else {
            cout << spisok2[0].y;
        }
    }
```

```
    return 0;
}
```

Task F (7)

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

using namespace std;

#define x first
#define y second
typedef long long ll;

ll n, k, ans = 0;
vector<pair <ll, ll> > spisok;

void perebor (ll d, ll a, ll b, ll cur_ans) {
    if (d == n) {
        ans = max(ans, cur_ans);
        return;
    }

    else if (d == -1) {
        perebor(d + 1, a + k, b, cur_ans);
        perebor(d + 1, a, b + k, cur_ans);
    } else {

        if (a > spisok[d].x) {
            cur_ans += spisok[d].x;
            a -= spisok[d].x;
        } else {
            cur_ans += a;
            a = 0;
        }

        if (b > spisok[d].y) {
            cur_ans += spisok[d].y;
            b -= spisok[d].y;
        } else {
            cur_ans += b;
            b = 0;
        }


        perebor(d + 1, a + k, b, cur_ans);
        perebor(d + 1, a, b + k, cur_ans);

    }

}

int main()
{
    ios::sync_with_stdio(false);
    cin.tie(NULL);

    ll a, b;

    cin >> n >> k;
    for (int i = 0; i < n; i++) {
        cin >> a >> b;
        spisok.push_back({a, b});
    }

    perebor(-1, 0, 0, 0);

    cout << ans;

    return 0;
}
```