

Олимпиада СПбГУ по информатике 2018/19 учебного года

Хорохорин Андрей Сергеевич

A	B	C	D	E	F	Sum
100	100	100	60	35	0	395

Task A (100)

```
#include <bits/stdc++.h>
using namespace std;

signed main() {
#ifdef MY_DEBUG
    freopen("../input.txt", "r", stdin);
    freopen("../output.txt", "w", stdout);
#else
    ios_base::sync_with_stdio(0); cin.tie(0);
#endif
    int n, m;
    cin >> n >> m;
    for (int i = 0; i < 10; ++i) {
        if ((double) m / pow(2, i) == n) {
            cout << "Yes";
            return 0;
        }
    }
    cout << "No";
    return 0;
}
```

Task B (100)

```
#include <bits/stdc++.h>
using namespace std;

signed main() {
#ifdef MY_DEBUG
    freopen("../input.txt", "r", stdin);
    freopen("../output.txt", "w", stdout);
#else
    ios_base::sync_with_stdio(0); cin.tie(0);
#endif
    int n;
    cin >> n;
    string s;
    cin >> s;
    for (int i = 0; i < n - 1; ++i) {
        if ((s[i] == 'o' && s[i + 1] == 'r') || (s[i] == 'r' && s[i + 1] == 'o')) {
            cout << "Yes";
            return 0;
        }
    }
    for (int i = 0; i < n - 2; ++i) {
        if (s[i] == 'o' && s[i + 2] == 'r') {
            cout << "Yes";
            return 0;
        }
    }
    cout << "No";
    return 0;
}
```

Task C (100)

```
#include <bits/stdc++.h>
using namespace std;

vector <vector <int>> > v;
vector <pair <pair <int, int>, pair <int, int>>> maxchild;
vector <int> childs;
vector <int> parent;

int dfs(int u, int pr) {
    parent[u] = pr;
    int ans = 0;
    pair <int, int> max1 = {0, 0}, max2 = {0, 0};
    for (int i : v[u]) {
        if (i != pr) {
            int t = dfs(i, u);
            ans += t;
            if (t > max1.first) {
                max2 = max1;
                max1 = {t, i};
            } else {
                if (t > max2.first) {
                    max2 = {t, i};
                }
            }
        }
    }

    maxchild[u] = {max1, max2};
    childs[u] = ans;
    return ans + 1;
}

int getans(int u) {
    int ans = 0;
    for (int i : v[u]) {
        if (i == parent[u]) {
            ans = max(childs[parent[u]] - childs[u], ans);
            childs[u] += childs[parent[u]] - childs[u];
        }
        else {
            ans = max(ans, childs[i] + 1);
        }
    }
    return ans + 1;
}

signed main() {
#ifdef MY_DEBUG
    freopen("../input.txt", "r", stdin);
    freopen("../output.txt", "w", stdout);
#else
    ios_base::sync_with_stdio(0); cin.tie(0);
#endif
    int n;
    cin >> n;
    v.resize(n);
    maxchild.resize(n);
    childs.resize(n);
    parent.resize(n);
    for (int i = 0; i < n - 1; ++i) {
        int fr, to;
        cin >> fr >> to;
        fr--;
        to--;
        v[fr].emplace_back(to);
    }
}
```

```

        v[to].emplace_back(fr);
    }
    dfs(0, -1);
    /*
    for (int i = 0; i < n; ++i) {
        cout << maxchild[i].first.first << ' ' << maxchild[i].first.second << ' ' << maxchild[i].
            second.first << ' ' << maxchild[i].second.second << '\n';
    }
    */

    for (int i = 0; i < n; ++i) {
        //cout << childs[i] << ' ';
        cout << getans(i) << '└';
    }

    return 0;
}

```

Task D (60)

```
#include <bits/stdc++.h>
using namespace std;

signed main() {
#ifdef MY_DEBUG
    freopen("../input.txt", "r", stdin);
    freopen("../output.txt", "w", stdout);
#else
    ios_base::sync_with_stdio(0); cin.tie(0);
#endif
    string s;
    cin >> s;
    int t, n, p;
    cin >> t >> n >> p;
    if (n == 3) {
        for (int test = 0; test < t; ++test) {
            if (s == "split") {
                string s;
                cin >> s;
                string h1 = "aaaaaaa", h2 = "aaaaaaa", h3 = "aaaaaaa";
                h1[0] = 'a';
                h2[0] = 'b';
                h3[0] = 'c';
                for (int i = 0; i < 6; ++i) {
                    h1[i + 1] = s[i];
                }
                for (int i = 3; i < 9; ++i) {
                    h2[i - 2] = s[i];
                }
                for (int i = 0; i < 3; ++i) {
                    h3[i + 4] = s[i];
                }
                for (int i = 6; i < 9; ++i) {
                    h3[i - 5] = s[i];
                }
                cout << h1 << ' ' << h2 << ' ' << h3 << '\n';
            } else {
                string h1, h2;
                cin >> h1 >> h2;
                if (h1[0] > h2[0])
                    swap(h1, h2);
                if (h1[0] == 'a' && h2[0] == 'b') {
                    cout << h1.substr(1, 6) << h2.substr(4, 3) << '\n';
                }
                if (h1[0] == 'a' && h2[0] == 'c') {
                    cout << h1.substr(1, 6) << h2.substr(1, 3) << '\n';
                }
                if (h1[0] == 'b' && h2[0] == 'c') {
                    cout << h2.substr(4, 3) << h1.substr(1, 6) << '\n';
                }
            }
        }
    }
    return 0;
}

if (n == 5) {
    for (int test = 0; test < t; ++test) {
        if (s == "split") {
            string s;
            cin >> s;
            string h1 = "aaaaaaa", h2 = "aaaaaaa", h3 = "aaaaaaa", h4 = "aaaaaaa", h5 = "
                aaaaaaa";
            h1[0] = 'a';
            h2[0] = 'b';
            h3[0] = 'c';
            h4[0] = 'd';
            h5[0] = 'f';
            for (int i = 0; i < 6; ++i) {
```

```

        h1[i + 1] = s[i];
    }
    for (int i = 3; i < 9; ++i) {
        h2[i - 2] = s[i];
    }
    for (int i = 0; i < 3; ++i) {
        h3[i + 4] = s[i];
    }
    for (int i = 6; i < 9; ++i) {
        h3[i - 5] = s[i];
    }
    for (int i = 0; i < 3; ++i) {
        h4[i + 4] = s[i];
    }
    for (int i = 6; i < 9; ++i) {
        h4[i - 5] = s[i];
    }
    h5[1] = s[3];
    h5[2] = s[4];
    h5[3] = s[5];

    cout << h1 << '\u' << h2 << '\u' << h3 << '\u' << h4 << '\u' << h5 << '\n';
} else {
    vector<string> h(3);
    cin >> h[0] >> h[1] >> h[2];
    sort(h.begin(), h.end());
    if (h[0][0] == 'a' && h[1][0] == 'b') {
        cout << h[0].substr(1, 6) << h[1].substr(4, 3) << '\n';
        continue;
    }
    if ((h[1][0] == 'c' || h[1][0] == 'd') && h[2][0] == 'f') {
        cout << h[1].substr(4, 3) << h[2][1] << h[2][2] << h[2][3] << h[1].substr(1,
            3) << '\n';
        continue;
    }
    if ((h[2][0] == 'c' || h[2][0] == 'd') && h[0][0] == 'a') {
        cout << h[0].substr(1, 6) << h[2].substr(1, 3) << '\n';
        continue;
    }
    if ((h[2][0] == 'c' || h[2][0] == 'd') && h[0][0] == 'b') {
        cout << h[2].substr(4, 3) << h[0].substr(1, 6) << '\n';
        continue;
    }
}
}
return 0;
}
else {
    cout << "Not_worked";
}
}

```

Task E (35)

```
#include <bits/stdc++.h>
using namespace std;
// #define double long double

struct line {
    double a, b, c;
    line(double _a, double _b, double _c) {
        a = _a;
        b = _b;
        c = _c;
    }
    line(pair<double, double> p1, pair<double, double> p2) {
        a = p2.second - p1.second;
        b = p1.first - p2.first;
        c = -p1.first * a + p2.second * b;
    }
};

double len(pair<double, double> a, pair<double, double> b) {
    return sqrt((a.first - b.first) * (a.first - b.first) + (a.second - b.second) * (a.second - b.second));
}

signed main() {
#ifdef MY_DEBUG
    freopen("../input.txt", "r", stdin);
    freopen("../output.txt", "w", stdout);
#else
    ios_base::sync_with_stdio(0); cin.tie(0);
#endif
    int n;
    cin >> n;
    pair<double, double> p1, p2;

    vector<pair<pair<double, double>, int>> points(n);
    for (int i = 0; i < n; ++i) {
        cin >> points[i].first.first >> points[i].first.second;
    }
    cin >> p1.first >> p1.second >> p2.first >> p2.second;
    for (int i = 0; i < n; ++i) {
        if (p1.first > p2.first)
            points[i].first.second = abs(points[i].first.second);
        else
            points[i].first.second = -abs(points[i].first.second);
        points[i].second = i + 1;
    }
    if (p1.first < p2.first)
        sort(points.rbegin(), points.rend());
    else
        sort(points.begin(), points.end());
    if (points[0].first == points[1].first)
        cout << -1;
    else
        cout << points[0].second;
}
```

Task F (—)