

Олимпиада СПбГУ по информатике 2018/19 учебного года

Лесник Алексей Алексеевич

A	B	C	D	E	F	Sum
100	100	100	60	100	0	460

Task A (100)

```
#define _CRT_SECURE_NO_WARNINGS
#include<iostream>
#include<vector>
#include<string>
#include<algorithm>
#include<cmath>

using namespace std;

int main() {
    //freopen("input.txt", "r", stdin);
    //freopen("output.txt", "w", stdout);
    ios_base::sync_with_stdio(NULL);
    cin.tie(NULL);
    long long n;
    cin >> n;
    long long m;
    cin >> m;
    while (n < m) n *= 2;
    if (n == m) cout << "Yes";
    else cout << "No";

}
```

Task B (100)

```
#define _CRT_SECURE_NO_WARNINGS
#include<iostream>
#include<vector>
#include<string>
#include<algorithm>
#include<cmath>

using namespace std;

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

Task C (100)

```
#define _CRT_SECURE_NO_WARNINGS
#include<iostream>
#include<vector>
#include<string>
#include<algorithm>
#include<cmath>

using namespace std;

vector<vector<int>>> g;
vector<int> siz, parent;
int n;

int dfss(int v, int p) {
    parent[v] = p;
    int ans = 1;
    for (int u : g[v]) {
        if (u != p) {
            ans += dfss(u, v);
        }
    }
    return siz[v] = ans;
}

int get_size(int u, int v) {
    if (parent[u] != v)
        return n - siz[v];
    else
        return siz[u];
}

int main() {
    //freopen("input.txt", "r", stdin);
    //freopen("output.txt", "w", stdout);
    ios_base::sync_with_stdio(NULL);
    cin.tie(NULL);
    cin >> n;
    g.resize(n);
    for (int i = 0; i < n - 1; i++) {
        int x, y;
        cin >> x >> y;
        x--, y--;
        g[x].push_back(y);
        g[y].push_back(x);
    }
    parent.resize(n);
    siz.resize(n);
    dfss(0, -1);
    vector<int> ans(n, 1);
    for (int i = 0; i < n; i++) {
        for (int u : g[i]) {
            ans[i] = max(ans[i], get_size(u, i) + 1);
        }
    }
    for (int i : ans) {
        cout << i << "_";
    }
}
```

Task D (60)

```
#define _CRT_SECURE_NO_WARNINGS
#include<iostream>
#include<vector>
#include<string>
#include<algorithm>
#include<cmath>

using namespace std;

string str(char a1, char a2, char a3, char a4, char a5, char a6, char a7, char a8, char a9) {
    string ans = { a1, a2, a3, a4, a5, a6, a7, a8, a9 };
    return ans;
}

int main() {
    //freopen("input.txt", "r", stdin);
    //freopen("output.txt", "w", stdout);
    ios_base::sync_with_stdio(NULL);
    cin.tie(NULL);
    string s;
    cin >> s;
    int n, k, p;
    cin >> n >> k >> p;
    k = k / 2 + k % 2;
    if (s == "split") {
        for (int j = 0; j < n; j++) {
            string in;
            cin >> in;
            if (k == 2) {
                cout << 'a';
                for (int i = 0; i < 6; i++)
                    cout << in[i];
                cout << "\n";
                cout << 'b';
                for (int i = 0; i < 3; i++)
                    cout << in[i];
                for (int i = 6; i < 9; i++)
                    cout << in[i];
                cout << "\n";
                cout << 'c';
                for (int i = 3; i < 9; i++)
                    cout << in[i];
                cout << "\n";
            }
            else {
                cout << 'a';
                for (int i = 0; i < 6; i++)
                    cout << in[i];
                cout << "\n";
                cout << 'a';
                for (int i = 0; i < 6; i++)
                    cout << in[i];
                cout << "\n";
                cout << 'b';
                for (int i = 0; i < 3; i++)
                    cout << in[i];
                for (int i = 6; i < 9; i++)
                    cout << in[i];
                cout << "\n";
                cout << 'c';
                for (int i = 3; i < 9; i++)
                    cout << in[i];
                cout << "\n";
                cout << 'c';
                for (int i = 3; i < 9; i++)
                    cout << in[i];
                cout << "\n";
            }
        }
    }
}
```

```

    }
else {
    for (int j = 0; j < n; j++) {
        if (j) cout << "\n";
        vector<string> in(k);
        for (int i = 0; i < k; i++)
            cin >> in[i];
        sort(in.begin(), in.end());
        bool ok = false;
        for (int a = 1; a < k && !ok; a++) {
            for (int b = 0; b < a && !ok; b++) {
                if (in[b][0] == 'a' && in[a][0] == 'b') ok = true, cout <<
                    str(in[b][1], in[b][2], in[b][3], in[b][4], in[b][5],
                        in[b][6], in[a][4], in[a][5], in[a][6]);
                if (in[b][0] == 'a' && in[a][0] == 'c') ok = true, cout <<
                    str(in[b][1], in[b][2], in[b][3], in[b][4], in[b][5],
                        in[b][6], in[a][4], in[a][5], in[a][6]);
                if (in[b][0] == 'b' && in[a][0] == 'c') ok = true, cout <<
                    str(in[b][1], in[b][2], in[b][3], in[a][1], in[a][2],
                        in[a][3], in[a][4], in[a][5], in[a][6]);
            }
        }
    }
}

```

Task E (100)

```
#define _CRT_SECURE_NO_WARNINGS
#include<iostream>
#include<vector>
#include<string>
#include<algorithm>
#include<cmath>

using namespace std;
long long Max = 1e+15;
//#define __int128 long long

pair<long long, long long> dist(long long x, long long y, pair<long long, long long> b) {
    return{ abs(x - b.first), abs(y - b.second) };
}

bool amensheb(pair<__int128, __int128> a, pair<__int128, __int128> b) {
    __int128 ai = a.first * a.first + a.second * a.second;
    __int128 bi = b.first * b.first + b.second * b.second;
    return ai < bi;
}

int main() {
    //freopen("input.txt", "r", stdin);
    //freopen("output.txt", "w", stdout);
    ios_base::sync_with_stdio(NULL);
    cin.tie(NULL);
    int n;
    cin >> n;
    vector<pair<long long, long long>> in(n);
    for (int i = 0; i < n; i++) {
        cin >> in[i].first >> in[i].second;
    }
    long long x1, y1, x2, y2;
    cin >> x1 >> y1 >> x2 >> y2;
    long long dx = x2 - x1, dy = y2 - y1;
    if (n == 2) {
        long long deltax = in[0].first - in[1].first, deltay = in[0].second - in[1].second;
        swap(deltax, deltay);
        deltax *= -1;
        if ((__int128)dx * deltay - (__int128)deltax * dy != 0) {
            if ((__int128)dx * deltay - (__int128)deltax * dy > 0) {
                cout << "1";
            }
            else {
                cout << "2";
            }
            return 0;
        }
    }
    long long k = Max;
    if (dx != 0) k = min(abs(Max / dx), k);
    if (dy != 0) k = min(abs(Max / dy), k);
    x2 += dx * k, y2 += dy * k;
    x1 -= dx * k, y1 -= dy * k;
    vector<int> ans;
    pair<long long, long long> mindist = { Max, Max };
    for (int i = 0; i < n; i++) {
        pair<long long, long long> distnow = dist(x2, y2, in[i]);
        if (!amensheb(distnow, mindist) && !amensheb(mindist, distnow)) ans.push_back(i);
        if (amensheb(distnow, mindist)) {
            mindist = distnow;
            ans.assign(1, i);
        }
    }
    vector<int> ans2;
    mindist = { 0, 0 };
```

```

for (int i : ans) {
    pair<long long, long long> distnow = dist(x1, y1, in[i]);
    if (!amensheb(distnow, mindist) && !amensheb(mindist, distnow)) ans2.push_back(i);
    if (amensheb(mindist, distnow)) {
        mindist = distnow;
        ans2.assign(1, i);
    }
}
if (ans2.size() == 1) {
    cout << ans2[0] + 1;
}
else cout << "-1";
}

```

Task F (0)

```
#define _CRT_SECURE_NO_WARNINGS
#include<iostream>
#include<vector>
#include<string>
#include<algorithm>
#include<cmath>

using namespace std;

const long long INF = 1e+17;

int main() {
    //freopen("input.txt", "r", stdin);
    //freopen("output.txt", "w", stdout);
    ios_base::sync_with_stdio(NULL);
    cin.tie(NULL);
    long long n, k;
    cin >> n >> k;
    vector<vector<vector<pair<long long, long long>>>> dp(n + 1,
        vector<vector<pair<long long, long long>>>>(n + 2,
            vector<pair<long long, long long>>>(n + 2, { -INF, -INF })));
    vector<pair<long long, long long>> in(n + 1, { 0, 0 });
    for (int i = 0; i < n; i++) {
        cin >> in[i].first >> in[i].second;
    }
    long long ans = 0;
    dp[0][0][0] = { 0, 0 };
    dp[0][1][0] = { min(k, in[0].first), 0 };
    dp[0][0][1] = { 0, min(k, in[0].second) };
    for (int i = 0; i < n; i++) {
        for (int j = 0; j < i + 2; j++) {
            for (int u = 0; u < i + 2; u++) {
                //cout << dp[i][j][u].first + dp[i][j][u].second << " ";
                ans = max(ans, dp[i][j][u].first + dp[i][j][u].second);
                if (dp[i + 1][j + 1][u].first + dp[i + 1][j + 1][u].second < dp[i][j][u].first + dp[i][j][u].second + min((j + 1) * k - dp[i][j][u].first, in[i + 1].first)) {
                    dp[i + 1][j + 1][u] = { dp[i][j][u].first + min((j + 1) * k - dp[i][j][u].first, in[i + 1].first), dp[i][j][u].second };
                }
                if (dp[i + 1][j][u + 1].first + dp[i + 1][j][u + 1].second < dp[i][j][u].first + dp[i][j][u].second + min((u + 1) * k - dp[i][j][u].second, in[i + 1].second)) {
                    dp[i + 1][j][u + 1] = { dp[i][j][u].first, dp[i][j][u].second + min((u + 1) * k - dp[i][j][u].second, in[i + 1].second) };
                }
            }
        }
        //cout << "\n";
    }
    //cout << "\n";
    cout << ans;
}
```