

Олимпиада СПбГУ по информатике 2019/20 учебного года

A	B	C	D	E	F	Sum
100	100	100	40	12	18	370

Task A ()

```
#include <vector>
#include <iostream>
#include <algorithm>

using namespace std;

#define int long long

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

    int n;
    cin >> n;
    cout << n - 1;
    return 0;
}
```

Task B ()

```
#include <vector>
#include <iostream>
#include <algorithm>
#include <math.h>

using namespace std;

#define int long long

struct p{
    double x, y;
};
bool operator < (const p&a, const p&b){
    if (a.x == b.x) return a.y < b.y;
    return a.x < b.x;
}
signed main(){
    ios_base::sync_with_stdio(0);
    cin.tie(0);
    cout.tie(0);

    int n;
    cin >> n;
    vector<p> lst;
    for (int i =0; i < n; i++){
        p x;
        cin >> x.x >> x.y;
        lst.push_back(x);
    }
    if (n == 3){
        double serx = (lst[1].x + lst[2].x) / 2;
        double sery = (lst[1].y + lst[2].y) / 2;
        double a = (serx - lst[0].x) * 2 + lst[0].x;
        double b = (sery - lst[0].y) * 2 + lst[0].y;
        cout << lst[0].x << " " << lst[0].y << "\n";
        cout << lst[2].x << " " << lst[2].y << "\n";
        cout << a << " " << b << "\n";
        cout << (lst[1].x - lst[0].x) * 2 + lst[0].x << " " << (lst[1].y - lst[0].y) * 2 + lst[0].y << "\n";
        cout << (lst[1].x - lst[2].x) * 2 + lst[2].x << " " << (lst[1].y - lst[2].y) * 2 + lst[2].y << "\n";
        cout << (lst[1].x - a) * 2 + a << " " << (lst[1].y - b) * 2 + b << "\n";
    }
    else{
        sort(lst.begin(), lst.end());
        int ind =1;
        double l = 0;
        for (int i= 1; i < 6; i++){
            double c = sqrt((lst[i].x - lst[0].x) * (lst[i].x - lst[0].x) + (lst[i].y - lst[0].y) * (lst[i].y - lst[0].y));
            if (c > l){
                l = c;
                ind = i;
            }
        }
        double a = (lst[ind].x + lst[0].x) / 2, b =(lst[ind].y + lst[0].y) / 2;
        cout << lst[0].x << " " << lst[0].y << "\n";
        cout << a << " " << b << "\n";
        cout << lst[1].x << " " << lst[1].y << "\n";
    }
    return 0;
}
```

Task C ()

```
#include <vector>
#include <iostream>
#include <algorithm>

using namespace std;

#define int long long
#define INF 1000000000 + 7

signed main(){
    ios_base::sync_with_stdio(0);
    cin.tie(0);
    cout.tie(0);
    string t;
    cin >> t;
    int n;
    cin >> n;
    vector<string> lst;
    for (int i = 0; i < n; i++){
        string s;
        cin >> s;
        lst.push_back(s);
    }
    int ans= 0;
    for (int k =0; k < n;k++){
        string cur = lst[k];
        int x= INF;
        for (int i = 0; i < cur.size(); i++){
            int c = 0;
            int j = 0;
            int ii = i;
            while (j != t.size()){
                while (j != t.size() && t[j] != cur[ii]){
                    j++;
                    c++;
                }
                if (j == t.size()) break;
                if (ii != cur.size()) ii++;
                j++;
            }
            x = min(x, c);
        }
        ans += x;
    }
    cout << ans;
    return 0;
}
```

Task D ()

```
#include <vector>
#include <iostream>
#include <algorithm>
#include <set>

using namespace std;

#define int long long
#define INF 1000000000 + 7

signed main(){
    ios_base::sync_with_stdio(0);
    cin.tie(0);
    cout.tie(0);
    int n, m;
    cin >> n >> m;
    int st, f;
    int x, y;
    cin >> x >> y;
    x--;
    y--;
    st = x * m + y;
    cin >> x >> y;
    x--;
    y--;
    f = x * m + y;
    vector<vector<pair<int, int>>> lst, c;
    lst.assign(n * m, {});
    for (int i = 0; i < n; i++){
        c.push_back({});
        for (int j = 0; j < m; j++){
            int a, b;
            cin >> a >> b;
            c[i].push_back({a, b});
        }
    }
    for (int i = 0; i < n * m; i++){
        for (int j = 0; j < n * m; j++){
            if (i != j){
                int x1 = i / m, y1 = i % m;
                int x2 = j / m, y2 = j % m;
                int cur = abs(c[x1][y1].first + x1 - x2) + abs(c[x1][y1].second + y1 - y2);
                lst[i].push_back({j, cur});
            }
        }
    }

    vector<int> d;
    d.assign(n * m, INF);
    set<pair<int, int>> q;
    d[st] = 0;
    q.insert({0, st});
    while (!q.empty()){
        int v = (*q.begin()).second;
        q.erase(q.begin());
        for (pair<int, int> to:lst[v]){
            if (d[to.first] > d[v] + to.second){
                d[to.first] = d[v] + to.second;
                q.insert({d[to.first], to.first});
            }
        }
    }
    cout << d[f];
    return 0;
}
```

Task E ()

```
#include <vector>
#include <iostream>
#include <algorithm>
#include <math.h>

using namespace std;

#define int long long

signed main(){
    ios_base::sync_with_stdio(0);
    cin.tie(0);
    cout.tie(0);
    int n, m, b;
    cin >> n >> m >> b;

    if (b == 1){
        int x, y;
        cin >> x >> y;
        x--;
        y--;
        cout << "?_0_0_" << n * 2 << "_" << m * 2 << endl;
        int s, t;
        cin >> s >> t;
        if (s == 0){
            cout << "!_" << 2 * n - x << "_" << 2 * m - y;
        }
        else{
            cout << "!_" << -x << "_" << -y;
        }
    }
    else{
        int x, y;
        cin >> x >> y;
        x--;
        y--;
        int a, b;
        cin >> a >> b;
        a--;
        b--;
        int s1 = a - x, s2 = b - y;
        vector<pair<int, int>> lst;
        vector<pair<int, int>> used;
        lst.push_back({0, 0});
        lst.push_back({10000, 10000});
        cout << "?_0_0_" << 10000 << "_" << 10000 << endl;
        int s, t;
        cin >> s >> t;
        used.push_back({s, t});
        cout << "?_" << 50000 << "_" << 50000 << "_" << 70000 << "_" << 70000 << endl;
        lst.push_back({50000, 50000});
        lst.push_back({70000, 70000});
        cin >> s >> t;
        used.push_back({s, t});
        int xx = lst.size();
        int c = 2;
        for (int i = 0; i < xx; i++){
            int bl = 1;
            //cerr << 1;
            for (int j = 0; j < used.size(); j++){
                if (used[j] == lst[i]) bl = 0;
            }
            //cerr << 2;
            if (bl){
                //cerr << 3;
                lst.push_back({lst[i].first + s1, lst[i].second + s2});
                //cerr << 4;
                c--;
                if (c == 0) break;
            }
        }
    }
}
```

```

cout << "?_ " << lst[lst.size() - 1].first << "_ " << lst[lst.size() - 1].second << "_ " <<
    lst[lst.size() - 2].first << "_ " << lst[lst.size() - 2].second << endl;
cin >> s >> t;
used.push_back({s, t});
for (int i = 0; i < lst.size() - 2; i++){
    int b1 = 1;
    for (int j = 0; j < used.size(); j++){
        if (used[j] == lst[i]) b1 = 0;

    }
    if (b1){
        if (used[used.size() - 1] != make_pair(lst[i].first + s1, lst[i].second + s2)){
            cout << "!_ " << lst[i].first - x << "_ " << lst[i].second - y << "\n";
            return 0;
        }
    }
}

}
return 0;
}

```

Task F ()

```
#include <vector>
#include <iostream>
#include <algorithm>
#include <math.h>

using namespace std;

#define int long long
int n = 1, m;
vector<vector<int>>> g;
vector<int> p, d;
vector<int> ans;
int BL;

void dfs(int st, int cl = 0){
    d[st] = cl;
    for (int i : g[st]){
        if (d[i] != -1) BL = 0;
        dfs(i, cl + 1);
    }
}

void ch(){
    g.assign(n, {});
    for (int i = 0; i < n - 1; i++){
        g[p[i]].push_back(i + 1);
    }
    d.assign(n, -1);
    BL = 1;
    dfs(0);

    int x = 0;
    for (int i = 0; i < n; i++){
        if (d[i] == -1) BL = 0;
    }
    if (!BL) return;
    for (int i = 0; i < n; i++){
        for (int j = i + 1; j < n; j++){
            int a = i, b = j;
            while (d[a] < d[b]){
                b = p[b - 1];
                x++;
            }
            while (d[a] > d[b]){
                a = p[a - 1];
                x++;
            }
            while (a != b){
                a = p[a - 1];
                x++;
                b = p[b - 1];
                x++;
            }
        }
    }
    ans[x - 1]++;
}

void gen(){
    if (p.size() == n - 1){
        ch();
        return;
    }
    for (int i = 0; i < n; i++){
        p.push_back(i);
        gen();
        p.pop_back();
    }
}
```

```

signed main(){
    ios_base::sync_with_stdio(0);
    cin.tie(0);
    cout.tie(0);
    cin >> n >> m;
    p.clear();
    ans.assign(m, 0);
    gen();
    for (int i:ans){
        cout << i << " ";
    }
    cout << "\n";

    return 0;
}

```