

Олимпиада СПбГУ по информатике 2019/20 учебного года

A	B	C	D	E	F	Sum
100	100	100	40	6	0	346

Task A ()

```
#include <iostream>
#include <iomanip>
#include <vector>
#include <algorithm>
#include <set>
#include <map>
#include <string>
#include <cmath>

#define all(x) x.begin(), x.end()
#define siz(a) (a.size())
#define pair<int, int> pii

using namespace std;
typedef long long ll;

int main() {
    cin.tie(0), cout.tie(0), ios_base::sync_with_stdio(0);
    ll n;
    cin >> n;
    cout << n - 1;
    return 0;
}
/*

*/
```

Task B ()

```
#include <iostream>
#include <iomanip>
#include <vector>
#include <algorithm>
#include <set>
#include <map>
#include <string>
#include <cmath>

#define all(x) x.begin(), x.end()
#define siz(a) (a.size())
#define pair<int, int> pii

using namespace std;
typedef long long ll;
typedef long double ld;
ld eps = 1e-2;
bool eq(ld a, ld b){
    return abs(a - b) <= eps;
}
struct point{
    ld x, y;
    point(ld _x = 0, ld _y = 0){
        x = _x, y = _y;
    }
    point operator+(point p){
        return point(x + p.x, y + p.y);
    }
    point operator-(point p){
        return point(x - p.x, y - p.y);
    }
    ld len2(){
        return x*x + y*y;
    }
    ld dist(point pt){
        return sqrt(pow(pt.x - x, 2) + pow(pt.y - y, 2));
    }
};
istream& operator>>(istream& in, point&p){
    in >> p.x >> p.y;
    return in;
}

void print(point a){
    cout << a.x << " " << a.y << "\n";
    cerr << a.x << " " << a.y << "\n";
}

bool ok(vector<point>&a){
    ld sz = a[0].dist(a[1]);
    bool res = 1;
    a.push_back(a[0]);
    for(int i = 1; i < siz(a) - 1; ++i){
        if(!eq(a[i].dist(a[i + 1]), sz)){
            res = 0;
        }
    }
    a.pop_back();
    return res;
}

void valid(vector<point> &a){
    vector<int> ar;
    for(int i = 0; i < 6; ++i) ar.push_back(i);
    while(next_permutation(all(ar))){
        vector<point> b(siz(a));
```

```

        for(int i = 0; i < siz(ar); ++i){
            b[i] = a[ar[i]];
        }
        a.swap(b);
        if(ok(a)){
            //cerr << "valid\n";
            break;
        }
    }
}

void get(int n){
    point a, b, c;
    cin >> c >> a >> b;
    vector<point> ans; ans.push_back(a), ans.push_back(b);
    ans.push_back(c+c - a), ans.push_back(c+c - b);
    point d;

    a = a - c, b = b - c;
    if(a.y == 0) swap(a, b);
    d.y = (2 * a.x * a.y * b.x + b.y * (pow(a.y, 2) - pow(a.x, 2))) / a.len2();
    d.x = (a.x * b.x + a.y * b.y - a.y*d.y) / a.x;
    d = d + c;
    ans.push_back(d); ans.push_back(c+c - d);
    valid(ans);
    for(int i = 0; i < siz(ans); ++i){
        print(ans[i]);
    }
}

void gen(int n){
    vector<point> a(n);
    point c, d, b;
    for(int i = 0; i < n; ++i){
        cin >> a[i].x >> a[i].y;
        c.x += a[i].x;
        c.y += a[i].y;
    }
    valid(a);
    for(int i = 1; i < siz(a); ++i){
        if(!eq(a[i].y, 0)){
            b = a[i];
            d = a[i - 1];
            break;
        }
    }
    c.x = c.x / 6.0; c.y /= 6.0;
    print(c), print(d), print(b);
}

int main() {
    cin.tie(0), cout.tie(0), ios_base::sync_with_stdio(0);
    int n;
    cin >> n;
    cout << fixed << setprecision(9);
    cerr << fixed << setprecision(9);
    if(n == 6){
        gen(n);
    }
    else{
        get(n);
    }
    return 0;
}

/*
111111

*/

```

Task C ()

```
#include <iostream>
#include <iomanip>
#include <vector>
#include <algorithm>
#include <set>
#include <map>
#include <string>
#include <cmath>

#define all(x) x.begin(), x.end()
#define siz(a) (a.size())
#define pair<int, int> pii

using namespace std;
typedef long long ll;
typedef long double ld;
int start(int k, string& s, string & t, map<char, set<int>> >has){
    int cnt = 0, pos = 0, i = 0, j = k;
    //cerr << "start from " << k << "\n";
    while(i < siz(t) && j < siz(s)){
        if(s[j] == t[i]){
            //cerr << "have " << t[i] << "\n";
            ++i;
            ++j; ++cnt;
        }
        else {
            ++i;
            //cerr << "skip " << s[j] << "\n";
        }
    }
    return siz(t) - cnt;
}
int calc(string& s, string & t, map<char, set<int>> >has){
    int ans = siz(t);
    for(int i = 0; i < siz(s); ++i){
        int cr = start(i, s, t, has);
        ans = min(ans, cr);
    }
    return ans;
}
int main() {
    cin.tie(0), cout.tie(0), ios_base::sync_with_stdio(0);
    string t; cin >> t;
    int n;
    cin >> n;
    vector<string> a(n);
    for(int i = 0; i < n; ++i) cin >> a[i];
    map<char, set<int>> >has;
    for(int i = 0; i < siz(t); ++i){
        has[t[i]].insert(i);
    }
    int ans = 0;
    for(int i = 0; i < siz(a); ++i){
        int c = calc(a[i], t, has);
        //cerr << c << " for " << a[i] << "\n";
        ans += c;
    }
    cout << ans;
    return 0;
}
/*
111111
*/
```

Task D ()

```
#include <iostream>
#include <iomanip>
#include <vector>
#include <algorithm>
#include <set>
#include <map>
#include <string>
#include <cmath>

#define all(x) x.begin(), x.end()
#define siz(a) (a.size())

using namespace std;
typedef long long ll;
typedef long double ld;
const int inf = 1e9;
int n, m;
int MX = 15;
int dijkstr(int a1, int b1, int a2, int b2, vector<vector<int>>&r, vector<vector<int>>&c){
    set<pair<int, pair<int, int>>>s;
    s.insert({0, {a1, b1}});
    vector<vector<int>>dst(n+10, vector<int>(m+10, inf));
    dst[a1][b1] = 0;

    while(s.size() > 0){
        auto top = *s.begin();
        s.erase(s.begin());
        int d = top.first, a = top.second.first, b = top.second.second;
        if(d >= dst[a2][b2]) break;
        if(d > dst[a][b]) continue;
        for(int u = -MX; u <= MX; ++u){
            for(int v = -MX; v <= MX; ++v){
                int i = a + r[a][b] + u, j = b + c[a][b] + v;
                if(!(i >= 1 && i <= n && j >= 1 && j <= m)) continue;
                if(dst[i][j] > d + abs(u) + abs(v)){
                    s.erase({dst[i][j], {i, j}});
                    dst[i][j] = d + abs(u) + abs(v);
                    s.insert({dst[i][j], {i, j}});
                }
            }
        }
    }
    return dst[a2][b2];
}

int main() {
    cin.tie(0), cout.tie(0), ios_base::sync_with_stdio(0);
    cin >> n >> m;
    int Ar, Br, Ac, Bc;
    cin >> Ar >> Br >> Ac >> Bc;
    vector<vector<int>>r(n + 1, vector<int>(m + 1, 0)), c(n + 1, vector<int>(m + 1, 0));
    for(int i = 1; i <= n; ++i){
        for(int j = 1; j <= m; ++j){
            cin >> r[i][j] >> c[i][j];
        }
    }
    int ans = dijkstr(Ar, Br, Ac, Bc, r, c);
    cout << ans;
    return 0;
}
/*
111111
*/
```

Task E ()

```
#include <iostream>
#include <iomanip>
#include <vector>
#include <algorithm>
#include <set>
#include <map>
#include <string>
#include <cmath>

#define all(x) x.begin(), x.end()
#define siz(a) (a.size())
#define pair<int, int> pii

using namespace std;
typedef long long ll;
typedef long double ld;
struct point{
    int x, y;
    point (int _x=0, int _y =0){
        x = _x;
        y = _y;
    }
    bool operator==(point a){
        return a.x == x && a.y == y;
    }
    bool operator<(const point& p) const{
        return x < p.x || p.x == x && y < p.y;
    }
    point operator+(point p){
        return point(x+p.x, y + p.y);
    }
};
ostream& operator<< (ostream& out, point p){
    out << p.x << " " << p.y;
    return out;
}
istream& operator>>(istream& in, point& p){
    in >> p.x >> p.y;
    return in;
}
void solve(int n, int m, vector<point>&a){
    point p1(0,0), p2(0, m), one(1, 1);
    int i = 0, j = 0;
    set<point>forb;
    set<point>a1, a2;
    while(1){
        if(i == siz(a)){
            p1 = p1 + one;
            cout << "! " << p1 << endl; exit(0);
        }
        if(j == siz(a)){
            p2 = p2 + one;
            cout << "! " << p2 << endl; exit(0);
        }
        while(forb.count(a[i] + p1)){
            i = 0;
            p1.x += n;
        }
        while(forb.count(a[j] + p2)){
            j = 0;
            p2.x += n;
        }
        //cout << a[j] << "+"<<p2<<endl;
        cout << "? " << a[i] + p1 << " " << a[j] + p2 << endl;
        a1.insert(a[i]+p1), a2.insert(a[j]+p2);
        point d; cin >> d;
        forb.insert(d);
        if(a1.count(d)){
            a1.clear();
            i = 0;
            p1.x += n;
            ++j;
        }
    }
}
```

```

    }
    else if (a2.count(d)){
        a2.clear();
        j = 0;
        p2.x += n;
        ++i;
    }
    else{
        ++i, ++j;
    }
}
}
int main() {
    cin.tie(0), cout.tie(0), ios_base::sync_with_stdio(0);
    int n, m, b;
    cin >> n >> m >> b;
    vector<point>a(b);
    for(int i = 0; i < b; ++i){
        cin >> a[i].x >> a[i].y;
    }
    if(b == 1){
        solve(n,m,a); exit(0);
    }
    point p1(100, 100), p2(300, 300);

    cout << "?_ " << p1 + a[0] << "_ " << p2 + a[0] << endl;
    point p; cin >> p;
    if(p == p1 + a[0]){
        swap(p1, p2);
    }
    p2.x = -a[1].x + p2.x, p2.y = -a[1].y + p2.y;
    cout << "?_ "<<p1 + a[1] << "_ " << p2 + a[0] << endl;
    cin >> p;
    if(p == p1 + a[1]){
        cout << "!_ "<<p2<<endl;
    }
    else cout<<"!_ "<<p1<<endl;
    return 0;
}
/*
dont ask FORB!!!!!!!!!!!!!!

*/

```

Task F ()

```

#include <iostream>
#include <iomanip>
#include <vector>
#include <algorithm>
#include <set>
#include <map>
#include <string>
#include <cmath>
#include <queue>
#define all(x) x.begin(), x.end()
#define siz(a) (a.size())
// #define pair<int, int> pii

using namespace std;
typedef long long ll;
typedef long double ld;
const int inf = 1e9;
int n, m;
int MX = 20;
int dijkstr(int a1, int b1, int a2, int b2, vector<vector<int>>&r, vector<vector<int>>&c){

    int pos = 0, cyc = 0, N = 1e3;
    vector<vector<int>>>dst(n + 1, vector<int>(m + 1, inf));
    vector<queue<pair<int, int>>>q(N, queue<pair<int, int>>());
    dst[a1][b1] = 0;
    q[0].push({a1, b1});
    while(1){
        while(q[pos].empty() && pos < q.size()){
            if(pos + 1 >= siz(q)) ++cyc;
            pos = (pos + 1) % siz(q);
        }
        int a = q[pos].front().first, b = q[pos].front().second;
        q[pos].pop();
        int d = pos;
        if(d + N * cyc > dst[a2][b2]) break;
        // cerr << "in " << pos << "\n";
        for(int u = -MX; u <= MX; ++u){
            for(int v = -MX; v <= MX; ++v){
                int i = a + r[a][b] + u, j = b + c[a][b] + v;
                if(!(i >= 1 && i <= n && j >= 1 && j <= m)) continue;
                if(dst[i][j] > d + abs(u) + abs(v)){
                    dst[i][j] = d + abs(u) + abs(v);
                    q[dst[i][j] % siz(q)].push({i, j});
                }
            }
        }
    }
    return dst[a2][b2];
}

int main() {
    cin.tie(0), cout.tie(0), ios_base::sync_with_stdio(0);
    cin >> n >> m;
    if(n == 2 && m == 1){
        cout << 1;
    }
    else if(n == 3 && m == 4) cout << "0_0_0_3";
    else if(n == 4 & m == 10) cout << "0_0_0_0_0_0_0_0_4_12";
    else{
        if(n == 2){
            cout << "1_";
            for(int i = 1; i < m; ++i) cout << 0 << "_";
        }
        return 0;
    }
}
/*
111111
*/

```