

# Олимпиада СПбГУ по информатике 2019/20 учебного года

A	B	C	D	E	F	Sum
100	100	100	40	45	0	385

## Task A ()

```
#include <bits/stdc++.h>
#define int long long
#define INF 1e18
#define all(x) x.begin(), x.end()
using namespace std;

signed main() {
    // freopen("input.txt", "r", stdin);
    ios_base::sync_with_stdio(0);
    cin.tie(0); cout.tie(0);

    int n;

    cin >> n;

    cout << n - 1;
    return 0;
}
```

## Task B ()

```
#include <bits/stdc++.h>
#define int long long
#define endl "\n"
#define INF 1e18
#define EPS 0.01
#define all(x) x.begin(), x.end()
using namespace std;

struct point{
    double x, y;

    point() {}
    point(double x, double y) : x(x), y(y) {}
};

double _sqrt(int x){
    double l = 0, r = 1e9;

    for(int i = 0; i < 100; i++){
        double mid = (l + r) / 2;

        if(mid * mid > x) r = mid;
        else l = mid;
    }

    return l;
}

double dist(point a, point b){
    return sqrt((a.x - b.x)*(a.x - b.x) + (a.y - b.y)*(a.y - b.y));
}

struct vec{
    double x, y;

    vec(point a, point b){
        x = b.x - a.x;
        y = b.y - a.y;
    }
};

point operator+(point p, vec v){
    return point(p.x + v.x, p.y + v.y);
}

signed main(){
    // freopen("input.txt", "r", stdin);
    ios_base::sync_with_stdio(0);
    cin.tie(0); cout.tie(0);

    int n;
    vector<point> mas;

    cin >> n;

    mas.resize(n);
    for(auto& i : mas) cin >> i.x >> i.y;
    cout << fixed << setprecision(10);

    if(n == 6){
        double a = INF;

        for(int i = 0; i < n; i++){
            for(int j = 0; j < n; j++){
                if(i == j) continue;
                a = min(a, dist(mas[i], mas[j]));
            }
        }

        // for(int i = 0; i < n; i++) cout << i << " and " << (i + 1) % n << " = " << dist(mas[i],
        // mas[(i + 1) % n]) << endl;
    }
}
```

```

    for(int i = 0; i < n; i++) {
        for (int j = 0; j < n; j++) {
            for (int k = 0; k < n; k++) {
                if (i == j || i == k || j == k) continue;
                if (dist(mas[j], mas[i]) - a > EPS) continue;
                if (dist(mas[j], mas[k]) - a > EPS) continue;

                cout << mas[i].x << " " << mas[i].y << endl;
                cout << mas[j].x << " " << mas[j].y << endl;
                cout << mas[k].x << " " << mas[k].y << endl;
                return 0;
            }
        }
    }
} else {
    point cent, temp((mas[0].x + mas[2].x) / 2, (mas[0].y + mas[2].y) / 2);
    cent = mas[1] + vec(mas[1], temp) + vec(mas[1], temp);

    mas.push_back(mas[0] + vec(mas[0], cent) + vec(mas[0], cent));
    mas.push_back(mas[1] + vec(mas[1], cent) + vec(mas[1], cent));
    mas.push_back(mas[2] + vec(mas[2], cent) + vec(mas[2], cent));

    for(auto i : mas){
        cout << i.x << " " << i.y << endl;
    }
}

return 0;
}

```

## Task C ()

```
#include <bits/stdc++.h>
#define int long long
#define INF 1e18
#define all(x) x.begin(),x.end()
using namespace std;

signed main(){
//    freopen("input.txt","r",stdin);
    ios_base::sync_with_stdio(0);
    cin.tie(0); cout.tie(0);

    int q, sum = 0;
    string t,s;

    cin >> t >> q;

    while(q--){
        cin >> s;

        int mn = t.size();

        for(int i = 0; i < s.size(); i++){
            int cur = t.size();

            for(int j = 0, ptr = i; j < t.size(); j++){
                if(ptr < s.size() && s[ptr] == t[j]){
                    cur--;
                    ptr++;
                }
            }

            mn = min(mn, cur);
        }

        sum += mn;
    }

    cout << sum;
    return 0;
}
```

## Task D ()

```
#include <bits/stdc++.h>
#define int long long
#define INF 1e18
#define pii pair<int,int>
#define all(x) x.begin(),x.end()
using namespace std;

int n,m, cost[1001][1001];
pii s,f, mas[1001][1001];
set<pair<int,pii>> q;
pii mask[] = {
    {1,1},
    {-1,-1},
    {-1,1},
    {1,-1},
    {0,1},
    {0,-1},
    {1,0},
    {-1,0},
};

bool inside(pii p){
    return p.first >= 0 && p.first < n && p.second >= 0 && p.second < m;
}

void dej(pii p){
    for(int i = 0 ; i < n; i++) {
        for (int j = 0; j < m; j++) {
            //pii _p = {p.first + m.first, p.second + m.second};
            if (i == p.first && j == p.second) continue;
            pii _p = {i, j};
            if (!inside(_p)) continue;

            int x = abs(_p.first - mas[p.first][p.second].first - p.first);
            int y = abs(_p.second - mas[p.first][p.second].second - p.second);

            if (cost[_p.first][_p.second] > x + y + cost[p.first][p.second]) {
                cost[_p.first][_p.second] = x + y + cost[p.first][p.second];
                q.insert({cost[_p.first][_p.second], _p});
            }
        }
    }
}

signed main(){
    // freopen("input.txt","r",stdin);
    ios_base::sync_with_stdio(0);
    cin.tie(0); cout.tie(0);

    cin >> n >> m >> s.first >> s.second >> f.first >> f.second;
    s.first--, s.second--;
    f.first--, f.second--;

    for(int i = 0; i < n; i++){
        for(int j = 0; j < m; j++){
            cost[i][j] = INF;
            cin >> mas[i][j].first >> mas[i][j].second;
        }
    }

    q.insert({0ll,s});
    cost[s.first][s.second] = 0;

    while(q.size()){
        pii fir = (*q.begin()).second;
        q.erase(q.begin());
        dej(fir);
    }

    cout << cost[f.first][f.second];
    return 0;
}
```

## Task E ()

```
#include <iostream>
#include <vector>
#include <set>
#include <algorithm>
#include <string>
#define int long long
#define INF 1e18
#define pii pair<int,int>
#define all(x) x.begin(),x.end()
using namespace std;

int n, m, b;
vector<pii> p;
vector<int> cnt;
set<int> alive;
set<int>::iterator it;

string add(int y) {
    string res = "";
    int ord = cnt[y];

    res += to_string(p[ord].first);
    res += "_";
    res += to_string(p[ord].second + y * m);
    res += "_";

    cnt[y]++;
    return res;
}

pii get() {
    pii res;
    cin >> res.first >> res.second;

    auto temp = alive.find(res.second / m);
    if(temp != alive.end()) {
        if(it == temp) {
            it++;
            if(it == alive.end()) it = alive.begin();
        }
        alive.erase(temp);
    }

    return res;
}

signed main() {
    // freopen("input.txt","r",stdin);
    ios_base::sync_with_stdio(0);
    cin.tie(0);
    cout.tie(0);

    cin >> n >> m >> b;

    cnt.resize(1 << b);
    p.resize(b);

    for(int i = 0; i < b; i++) {
        cin >> p[i].first >> p[i].second;
        p[i].first--;
        p[i].second--;
    }

    for(int i = 0; i < (1 << b); i++) {
        alive.insert(i);
    }

    it = alive.begin();

    while(1) {
        int win = 0;
        string str = "?_";
```

```

    for(int i = 0; i < 2; i++) {
        if(cnt[*it] == b) {
            win = 1;
            break;
        }

        str += add(*it);

        it++;
        if(it == alive.end()) it = alive.begin();
    }

    if(win) break;
    cout << str << endl;
    get();
}

cout << "! " << 0 << " " << *it * m << endl;
return 0;
}

```

## Task F ()