

# Олимпиада СПбГУ по информатике 2019/20 учебного года

A	B	C	D	E	F	Sum
100	100	100	10	100	27	437

## Task A ()

```
#define _CRT_SECURE_NO_WARNINGS
#include <iostream>
#include <algorithm>
#include <vector>
#include <string>
#include <set>
#include <map>
#include <unordered_map>
#include <unordered_set>
#include <cmath>
#include <iomanip>
#include <ctime>
#include <random>
#include <chrono>

#pragma GCC optimize("Ofast")
#pragma GCC target("sse")

#define mp make_pair
#define pb push_back
using namespace std;
typedef long long ll;
typedef long double ld;

mt19937 gen(43);

int n;

int main()
{
#ifdef _DEBUG
    freopen("input.txt", "r", stdin);
    freopen("output.txt", "w", stdout);
#endif
    ios_base::sync_with_stdio(false);
    cin.tie(0);

    ll res = 1;
    cin >> n;
    cout << n - 1;
    return 0;
}
```

## Task B ()

```
#define _CRT_SECURE_NO_WARNINGS
#include <iostream>
#include <algorithm>
#include <vector>
#include <string>
#include <set>
#include <map>
#include <unordered_map>
#include <unordered_set>
#include <cmath>
#include <iomanip>
#include <ctime>
#include <random>
#include <chrono>

#pragma GCC optimize("Ofast")
#pragma GCC target("sse")

#define mp make_pair
#define pb push_back
using namespace std;
typedef long long ll;
typedef long double ld;

mt19937 gen(43);

struct pt
{
    ld x, y;
    pt(){}
    pt(const ld& _x, const ld& _y)
    {
        x = _x;
        y = _y;
    }
    pt operator - (const pt& a) const
    {
        return pt(x - a.x, y - a.y);
    }
    bool operator == (const pt& a) const
    {
        return x == a.x && y == a.y;
    }
};

int k;
pt a[6], X;

ld vect(const pt& a, const pt& b)
{
    return a.x * b.y - a.y * b.x;
}

bool cmp(const pt& a, const pt& b)
{
    if(a == X)
        return true;
    if(b == X)
        return false;
    return vect(b - X, a - X) < 0;
}

bool ch(vector<pt>& a)
{
    a.pb(a[0]);
    bool ch = true;
    for(int i = 0; i < 5; i++)
        ch &= vect(a[i + 2] - a[i + 1], a[i + 1] - a[i]) < 0;
    a.pop_back();
    return ch;
}
```

```

int main()
{
#ifdef _DEBUG
    freopen("input.txt", "r", stdin);
    freopen("output.txt", "w", stdout);
#endif
    ios_base::sync_with_stdio(false);
    cin.tie(0);
    cin >> k;
    cout << fixed << setprecision(10);
    if(k == 6)
    {
        X.x = X.y = 1e9;
        for(int i = 0; i < 6; i++)
        {
            cin >> a[i].x >> a[i].y;
            if(X.y > a[i].y)
                X = a[i];
            else
                if(X.y == a[i].y && X.x > a[i].x)
                    X = a[i];
        }
        sort(a, a + 6, cmp);
        for(int i = 0; i < 6; i += 2)
            cout << a[i].x << ' ' << a[i].y << '\n';
    }
    if(k == 3)
    {
        pt A, B, C;
        cin >> A.x >> A.y >> B.x >> B.y >> C.x >> C.y;
        pt mid = pt((B.x + C.x) / 2, (B.y + C.y) / 2);
        pt centr = pt(ld(2) / ld(3) * mid.x + A.x / ld(3), ld(2) / ld(3) * mid.y + A.y /
            ld(3));
        cout << A.x << ' ' << A.y << '\n';
        cout << ld(2) * centr.x - B.x << ' ' << ld(2) * centr.y - B.y << '\n';
        cout << C.x << ' ' << C.y << '\n';
        cout << ld(2) * centr.x - A.x << ' ' << ld(2) * centr.y - A.y << '\n';
        cout << B.x << ' ' << B.y << '\n';
        cout << ld(2) * centr.x - C.x << ' ' << ld(2) * centr.y - C.y << '\n';
    }
    return 0;
}

```

## Task C ()

```
#define _CRT_SECURE_NO_WARNINGS
#include <iostream>
#include <algorithm>
#include <vector>
#include <string>
#include <set>
#include <map>
#include <unordered_map>
#include <unordered_set>
#include <cmath>
#include <iomanip>
#include <ctime>
#include <random>
#include <chrono>

#pragma GCC optimize("Ofast")
#pragma GCC target("sse")

#define mp make_pair
#define pb push_back
using namespace std;
typedef long long ll;
typedef long double ld;

mt19937 gen(43);

string t, s;
int n;

int f(const string& s, const string& t)
{
    int res = 0;
    for(int beg = 0; beg < s.length(); beg++)
    {
        int i = beg, j = 0, found = 0;
        while(j < t.length() && i < s.length())
        {
            while(j < t.length() && s[i] != t[j])
                j++;
            if(j < t.length() && s[i] == t[j])
            {
                found++;
                i++;
                j++;
            }
            else
                break;
        }
        res = max(res, found);
    }
    return t.length() - res;
}

int main()
{
#ifdef _DEBUG
    freopen("input.txt", "r", stdin);
    freopen("output.txt", "w", stdout);
#endif
    ios_base::sync_with_stdio(false);
    cin.tie(0);
    cin >> t >> n;
    int ans = 0;
    while(n--)
    {
        cin >> s;
        ans += f(s, t);
    }
    cout << ans;
    return 0;
}
```

## Task D ()

```
#define _CRT_SECURE_NO_WARNINGS
#include <iostream>
#include <algorithm>
#include <vector>
#include <string>
#include <set>
#include <stack>
#include <deque>
#include <queue>
#include <map>
#include <unordered_map>
#include <unordered_set>
#include <cmath>
#include <iomanip>
#include <cassert>
#include <ctime>
#include <random>
#include <chrono>

#pragma GCC optimize("Ofast")
#pragma GCC target("sse")

#define mp make_pair
#define pb push_back
using namespace std;
typedef long long ll;
typedef long double ld;

mt19937 gen(43);

int n, m, sx, sy, tx, ty, r[1000][1000], c[1000][1000];
int d[1000][1000];
bool used[1010][1010];

bool dfs(int i, int j)
{
    used[i][j] = true;
    if(i == tx && j == ty)
        return true;
    if(i == -1 || i == n || j == -1 || j == m)
        return false;
    if(used[i + r[i][j], j + c[i][j]])
        if(d[i + r[i][j]][j + c[i][j]] == 0)
        {
            d[i][j] = 0;
            return true;
        }
    else
        return false;
    else
        if(dfs(i + r[i][j], j + c[i][j]))
        {
            d[i][j] = 0;
            return true;
        }
    else
        return false;
}

int change(int ty, int sy)
{
    if(ty > sy)
        return -1;
    return 1;
}

int rasst(int i1, int j1, int i2, int j2)
{
    int vx = i2 - i1;
    if(vx > 0)
    {
        if(r[i1][j1] == 1)
```

```

        vx--;
        if(r[i1][j1] == -1)
            vx++;
    }
    else
        if(vx == 0)
        {
            if(r[i1][j1] != 0)
                vx = 1;
        }
        else
            if(vx < 0)
            {
                if(r[i1][j1] == 1)
                    vx++;
                if(r[i1][j1] == -1)
                    vx--;
            }
    }
    int vy = j2 - j1;
    if(vy > 0)
    {
        if(c[i1][j1] == 1)
            vy--;
        if(c[i1][j1] == -1)
            vy++;
    }
    else
        if(vy == 0)
        {
            if(c[i1][j1] != 0)
                vy = 1;
        }
        else
            if(vy < 0)
            {
                if(c[i1][j1] == 1)
                    vy++;
                if(c[i1][j1] == -1)
                    vy--;
            }
    }
    return abs(vx) + abs(vy);
}

int main()
{
#ifdef _DEBUG
    freopen("input.txt", "r", stdin);
    freopen("output.txt", "w", stdout);
#endif
    ios_base::sync_with_stdio(false);
    cin.tie(0);
    cin >> n >> m >> sx >> sy >> tx >> ty;
    sx--;
    sy--;
    tx--;
    ty--;
    for(int i = 0; i < n; i++)
        for(int j = 0; j < m; j++)
            cin >> r[i][j] >> c[i][j];

    if(n == 1)
    {
        /*
        int dp[1000];
        dp[ty] = 0;*/
        int ans = abs(ty - sy);
        if(ty > sy)
        {
            for(int i = sy; i < ty; i++)
                ans -= (c[0][i] == 1);
            ans += (c[0][sy] == -1 && (sy == 0 || c[0][sy - 1] != 1));
            /*for(int i = ty - 1; i >= sy; i--)
            {
                dp[i] = 1e9;
                for(int j = i + 1; j <= ty; j++)

```

```

                                dp[i] = min(dp[i], dp[j] + rasst(0, i, 0, j));
                                }*/
                                }
                                if (ty < sy)
                                {
                                    for (int i = sy; i > ty; i--)
                                        ans -= (c[0][i] == -1);
                                    ans += (c[0][sy] == 1 && (sy == m - 1 || c[0][sy + 1] != -1));
                                    /*for (int i = ty + 1; i <= sy; i++)
                                    {
                                        dp[i] = 1e9;
                                        for (int j = i - 1; j >= ty; j--)
                                            dp[i] = min(dp[i], dp[j] + rasst(0, i, 0, j));
                                    }
                                    */
                                    //cout << dp[sy] << '\n';
                                    cout << ans;
                                    return 0;
                                }
                                if (n == 3)
                                    if (m == 3)
                                        cout << 1;
                                    else
                                        cout << 4;
                                /*for (int i = 0; i < n; i++)
                                    for (int j = 0; j < m; j++)
                                        d[i][j] = 1e9;
                                d[tx][ty] = 0;
                                for (int i = 0; i < n; i++)
                                    for (int j = 0; j < m; j++)
                                        if (!used[i][j])
                                            dfs(i, j);
                                set <pair <int, pair <int, int>>> s;
                                for (int i = 0; i < n; i++)
                                    for (int j = 0; j < m; j++)
                                        if (d[i][j] == 0)
                                            s.insert(mp(0, mp(i, j)));*/

                                return 0;
                            }
    }
}

```

## Task E ()

```
#define _CRT_SECURE_NO_WARNINGS
#include <iostream>
#include <algorithm>
#include <vector>
#include <string>
#include <set>
#include <map>
#include <unordered_map>
#include <unordered_set>
#include <cmath>
#include <iomanip>
#include <cassert>
#include <ctime>
#include <random>
#include <chrono>

#pragma GCC optimize("Ofast")
#pragma GCC target("sse")

#define mp make_pair
#define pb push_back
using namespace std;
typedef long long ll;
typedef long double ld;

mt19937 gen(43);

int n, m, B;
set<int> good;
set<int> bad;
pair<ll, ll> beg[10000], p[15];

void perform(const pair<ll, ll>& a, const pair<ll, ll>& b)
{
    cout << "?_ " << a.first << '_ ' << a.second << '_ ' << b.first << '_ ' << b.second << endl;
    pair<ll, ll> v;
    cin >> v.first >> v.second;
    for(auto it = good.begin(); it != good.end(); it++)
        if(v.first >= beg[*it].first && v.first <= beg[*it].first + n - 1)
            if(v.second >= beg[*it].second && v.second <= beg[*it].second + m - 1)
            {
                bad.insert(*it);
                break;
            }
}

void rec(int iter)//set good,
{
    bad.clear();
    if(iter == B)
    {
        cout << "!_ " << beg[*good.begin()].first << '_ ' << beg[*good.begin()].second << endl;
        exit(0);
    }
    int need = (1 << (B - iter));
    while(good.size() > need)
        good.erase(good.begin());
    for(auto it = good.begin(); it != good.end(); it++)
    {
        pair<ll, ll> a, b;
        a = mp(beg[*it].first + p[iter].first - 1, beg[*it].second + p[iter].second - 1);
        it++;
        b = mp(beg[*it].first + p[iter].first - 1, beg[*it].second + p[iter].second - 1);
        perform(a, b);
    }
    for(auto it = bad.begin(); it != bad.end(); it++)
        good.erase(good.find(*it));
    rec(iter + 1);
}

int main()
```



```

{
#ifdef _DEBUG
//freopen("input.txt", "r", stdin);
//freopen("output.txt", "w", stdout);
#endif
ios_base::sync_with_stdio(false);
cin.tie(0);
cin >> n >> m >> B;
for(int i = 0; i < B; i++)
    cin >> p[i].first >> p[i].second;
int CNT = (1 << B) - 1;
for(ll beg_x = 0; CNT >= 0 && beg_x + n - 1 <= 1e18; beg_x += 2 * n)
    for(ll beg_y = 0; CNT >= 0 && beg_y + m - 1 <= 1e18; beg_y += 2 * m)
    {
        beg[CNT] = mp(beg_x, beg_y);
        good.insert(CNT);
        CNT--;
    }
rec(0);
return 0;
}

```

## Task F ()

```
#define _CRT_SECURE_NO_WARNINGS
#include <iostream>
#include <algorithm>
#include <vector>
#include <string>
#include <set>
#include <stack>
#include <deque>
#include <queue>
#include <map>
#include <unordered_map>
#include <unordered_set>
#include <cmath>
#include <iomanip>
#include <cassert>
#include <ctime>
#include <random>
#include <chrono>

#pragma GCC optimize("Ofast")
#pragma GCC target("sse")

#define mp make_pair
#define pb push_back
using namespace std;
typedef long long ll;
typedef long double ld;

mt19937 gen(43);

int n, m;
ll dp[30 * 30 * 5];
vector <pair <int, int>> ed;
set <vector <pair <int, int>>> s;
vector <vector <int>> gr;
bool used[100];

void dfs_component(int v)
{
    used[v] = true;
    for(int i = 0; i < gr[v].size(); i++)
        if(!used[gr[v][i]])
            dfs_component(gr[v][i]);
}

void dfs(int v, int pr, ll d, ll& ans)
{
    ans += d;
    for(int i = 0; i < gr[v].size(); i++)
        if(gr[v][i] != pr)
            dfs(gr[v][i], v, d + 1, ans);
}

void check(vector <pair <int, int>>& ed)
{
    if(s.find(ed) != s.end())
        return;
    gr.clear();
    gr.resize(n);
    for(int i = 0; i < ed.size(); i++)
    {
        gr[ed[i].first].pb(ed[i].second);
        gr[ed[i].second].pb(ed[i].first);
    }
    for(int i = 0; i < n; i++)
        used[i] = 0;
    dfs_component(0);
    bool ch = true;
    for(int i = 0; i < n; i++)
        ch &= (used[i]);
    if(!ch)
        return;
```



```

        rec(0, sos);
        //cout << "CASE " << n << ":\n";
        for(int i = 1; i <= m; i++)
            cout << dp[i] << '⌒';
        //cout << "\n\n";
        //cerr << "CASE " << n << "PASSED!\n";
    }
    return 0;
}

```