

Олимпиада СПбГУ по информатике 2019/20 учебного года

A	B	C	D	E	F	Sum
100	100	100	40	45	14	399

Task A ()

```
#include <bits/stdc++.h>

using namespace std;

int main()
{
#ifdef MY
    freopen("input.txt", "r", stdin);
    freopen("output.txt", "w", stdout);
#endif // MY
    int n;
    cin >> n;
    int a = 0, b = 0;
    int res = 0;
    for (int i = 0; i < 100; i++) {
        int cnt = n + a + b;
        a = cnt / 2;
        b = 0;
        res = max(res, a);
    }
    cout << res;
    return 0;
}
```

Task B ()

```
#include <bits/stdc++.h>

using namespace std;

struct p{
    double x, y;
    p(){}
    p(double _x, double _y) {
        x = _x;
        y = _y;
    }
    void scan() {
        cin >> x >> y;
    }
    void print() {
        cout << x << " " << y << endl;
    }
    bool operator < (p & a) {
        if(x != a.x)
            return x < a.x;
        return y < a.y;
    }
};

p get(p a, p b, p c) {
    p d;
    d.x = a.x + b.x;
    d.y = a.y + b.y;
    d.x /= 2.0;
    d.y /= 2.0;
    d.x = d.x + (d.x - c.x) / 3.0;
    d.y = d.y + (d.y - c.y) / 3.0;
    return d;
}

p A[6];
double v(const p & a, const p & b) {
    return a.x * b.y - a.y * b.x;
}

bool cmp(const p a, const p b) {
    return (v(p(a.x - A[0].x, a.y - A[0].y), p(b.x - A[0].x, b.y - A[0].y)) > 0);
}

int main()
{
#ifdef MY
    freopen("input.txt", "r", stdin);
    freopen("output.txt", "w", stdout);
#endif // MY
    ios_base::sync_with_stdio(false);
    cin.tie(0);
    cout << fixed << setprecision(5);
    int n;
    cin >> n;
    if(n == 6) {
        for(int i = 0; i < 6; i++)
            A[i].scan();
        for(int i = 1; i < 6; i++) {
            if(A[i] < A[0])
                swap(A[i], A[0]);
        }
        sort(A + 1, A + 6, &cmp);
        A[0].print();
        A[2].print();
        A[4].print();
        return 0;
    }
    if(n == 3) {
        p a[6];
        a[0].scan();
        a[2].scan();
        a[4].scan();
        a[1] = get(a[0], a[2], a[4]);
    }
}
```

```
    a[3] = get(a[2], a[4], a[0]);  
    a[5] = get(a[4], a[0], a[2]);  
    for(int i = 0; i < 6; i++) {  
        a[i].print();  
    }  
    return 0;  
}  
return 0;  
}
```

Task C ()

```
#include <bits/stdc++.h>

using namespace std;

struct p{
    double x, y;
    p(){}
    p(double _x, double _y) {
        x = _x;
        y = _y;
    }
    void scan() {
        cin >> x >> y;
    }
    void print() {
        cout << x << " " << y << endl;
    }
    bool operator < (p & a) {
        if(x != a.x)
            return x < a.x;
        return y < a.y;
    }
};

p get(p a, p b, p c) {
    p d;
    d.x = a.x + b.x;
    d.y = a.y + b.y;
    d.x /= 2.0;
    d.y /= 2.0;
    d.x = d.x + (d.x - c.x) / 3.0;
    d.y = d.y + (d.y - c.y) / 3.0;
    return d;
}

p A[6];
double v(const p & a, const p & b) {
    return a.x * b.y - a.y * b.x;
}

bool cmp(const p a, const p b) {
    return (v(p(a.x - A[0].x, a.y - A[0].y), p(b.x - A[0].x, b.y - A[0].y)) > 0);
}

string t;
int dp[10001][501];

int get(string s) {
    //cout << t;
    for(int i = 0; i <= s.size(); i++) {
        for(int j = 0; j <= t.size(); j++)
            dp[i][j] = -1;
    }
    for(int i = 0; i < s.size(); i++)
        dp[i][0] = 0;
    for(int i = 1; i <= s.size(); i++) {
        for(int j = 1; j <= t.size(); j++) {
            //cout << i << " " << j << " " << s[i - 1] << " " << t[j - 1] << endl;
            if(s[i - 1] != t[j - 1])
                continue;
            dp[i][j] = j - 1;
            for(int k = 0; k < j; k++) {
                if(dp[i - 1][k] == -1)
                    continue;
                dp[i][j] = min(dp[i][j], dp[i - 1][k] + j - k - 1);
            }
        }
    }
    int res = t.size();
    for(int i = 0; i <= s.size(); i++) {
        for(int j = 0; j <= t.size(); j++) {
            //cout << i << " " << j << " " << dp[i][j] << endl;
            if(dp[i][j] != -1)
```

```

        res = min(res, dp[i][j] + (int)t.size() - j);
    }
}
return res;
}

int main()
{
#ifdef MY
    freopen("input.txt", "r", stdin);
    freopen("output.txt", "w", stdout);
#endif // MY
    ios_base::sync_with_stdio(false);
    cin.tie(0);
    cout << fixed << setprecision(5);

    cin >> t;
    int n;
    cin >> n;
    vector<string> s(n);
    int res = 0;
    for(int i = 0; i < n; i++) {
        cin >> s[i];
        res += get(s[i]);
    }
    cout << res;
    return 0;
}

```

Task D ()

```
#include <bits/stdc++.h>

using namespace std;

mt19937 gen(41);

#define ll long long

int n, m;

pair<int, int> s, t;

pair<int, int> a[1000][1000];

int h(const pair<pair<int, int>, pair<int, int>> &a) {
    return a.first.first * 60 + a.first.second * 60*60 + a.second.first*60*60*60 + a.second.second
        * 60 *60*60*60;
}

int H(const pair<int, int> &a) {
    return a.first * 60 + a.second * 60*60;
}

unordered_map<int, int> d;
unordered_map<int, int> len;
bool used[1000][1000];
int main() {
#ifdef MY
    freopen("input.txt", "r", stdin);
    freopen("output.txt", "w", stdout);
#endif // MY
    ios_base::sync_with_stdio(false);
    cin.tie(0);
    cout << fixed << setprecision(5);
    d.max_load_factor(0.5);
    d.reserve(50*50*50*50);
    len.max_load_factor(0.5);
    len.reserve(50*50);
    cin >> n >> m;
    cin >> s.first >> s.second >> t.first >> t.second;
    s.first--;
    s.second--;
    t.first--;
    t.second--;
    for(int i = 0; i < n; i++) {
        for(int j = 0; j < m; j++) {
            cin >> a[i][j].first >> a[i][j].second;
        }
    }
    for(int i = 0; i < n; i++) {
        for(int j = 0; j < m; j++) {
            for(int k = 0; k < n; k++) {
                for(int l = 0; l < m; l++) {
                    d[h({{i, j}, {k, l}})] = abs(k - i - a[i][j].first) + abs(l - j - a[i][j].second);
                    //cout << d[{{i, j}, {k, l}}] << endl;
                }
            }
        }
    }
    for(int i = 0; i < n; i++) {
        for(int j = 0; j < m; j++) {
            len[H({i, j})] = 1e9;
        }
    }
    set<pair<int, pair<int, int>>> q;
    q.insert({0, s});
    len[H(s)] = 0;
    used[s.first][s.second] = true;
    while(!q.empty()) {
        auto v = q.begin()->second;
        if(v == t) {
```

```

        break;
    }
    q.erase(q.begin());
    for(int i = 0; i < n; i++) {
        for(int j = 0; j < m; j++) {
            if(used[i][j])
                continue;
            if(len[H({i, j})] > len[H(v)] + d[h({v, {i, j}})]) {
                q.erase({len[H({i, j})], {i, j}});
                len[H({i, j})] = len[H(v)] + d[h({v, {i, j}})];
                q.insert({len[H({i, j})], {i, j}});
            }
        }
    }
}
for(int i = 0; i < n; i++) {
    for(int j = 0; j < m; j++) {
        //cout << len[{i, j}] << " ";
    }
    //cout << endl;
}
cout << len[H(t)];
return 0;
}

```

Task E ()

```
#include <bits/stdc++.h>

using namespace std;

mt19937 gen(41);

#define ll long long

ll n, m, B;

vector<pair<ll, ll>> a;

struct pole{
    ll i, j, cnt = 0;
    bool f = true;
    pole() {}
    pole(ll _i, ll _j, ll _cnt) {
        i = _i;
        j = _j;
        cnt = _cnt;
    }
};

vector<pole> b;

ll cnt_p = 0;

void add_pole() {
    b.push_back(pole(0, m * cnt_p++, 0));
}

void del(ll x, ll y) {
    b[(y - 1) / m].f = false;
}

void kek(ll ind) {
    cout << a[b[ind].cnt].first << " " << a[b[ind].cnt].second + ind * m;
    b[ind].cnt++;
}

ll lol = 0;
bool hod() {
    lol++;
    ll ind1 = -1;
    ll ind2 = -1;
    if(b.size() == 0 || lol < 1000) {
        ind1 = cnt_p;
        add_pole();
        ind2 = cnt_p;
        add_pole();
    }
    else {
        int minx = 100000;
        for(int i = 0; i < b.size(); i++) {
            int j = gen() % b.size();
            if(b[i].f && b[i].cnt < minx) {
                minx = b[i].cnt;
                ind1 = i;
            }
        }
        if(ind1 == -1) {
            ind1 = cnt_p;
            add_pole();
        }
        else {
            if(gen() % 10000 == 0) {
                ind1 = cnt_p;
                add_pole();
            }
        }
        ind2 = -1;
        minx = 100000;
        for(int i = 0; i < b.size(); i++) {
```



```

        int j = gen() % b.size();
        if(b[i].f && b[i].cnt < minx && i != ind1) {
            minx = b[i].cnt;
            ind2 = i;
        }
    }
    if(ind2 == -1) {
        ind2 = cnt_p;
        add_pole();
    }
    else {
        if(gen() % 10000 == 0) {
            ind2 = cnt_p;
            add_pole();
        }
    }
}
cout << "?_";

kek(ind1);
cout << "_";
kek(ind2);
cout << endl;
ll x, y;
cin >> x >> y;
del(x, y);
ll maxc = 0;
for(int i = 0; i < b.size(); i++) {
    if(b[i].f)
        maxc = max(maxc, b[i].cnt);
    //cout << b[i].f << " " << b[i].cnt;
    if(b[i].f && b[i].cnt == B) {
        cout << "!_" << 1 << "_" << i * m + 1 << endl;
        return false;
    }
}
return true;
//cerr << maxc << endl;
}

int main() {
#ifdef MY
    //freopen("input.txt", "r", stdin);
    //freopen("output.txt", "w", stdout);
#endif // MY
    //ios_base::sync_with_stdio(false);
    //cin.tie(0);
    //cout << fixed << setprecision(5);
    cin >> n >> m >> B;
    a.resize(B);
    for(ll i = 0; i < B; i++) {
        cin >> a[i].first >> a[i].second;
    }
    while(hod()) {
        assert(lol < 8000);
    }
    return 0;
}

```

Task F ()

```
#include <bits/stdc++.h>

using namespace std;

mt19937 gen(41);

#define ll long long

vector<int> son[30];
vector<vector<int>>> g;
ll kek[1000000];

int n, m;

ll cur = 0;
unordered_set<int> s;
ll mod = 1e9 + 7;
ll P = 29, K = 41;

int get_h(vector<vector<int>>> &a) {
    ll h = 0;
    for(int i = 0; i < n; i++) {
        h *= K;
        h %= mod;
        for(auto to : g[i]) {
            h *= P;
            h += to + 1;
            h %= mod;
        }
    }
    return h;
}

void dfs(int v, int p = -1, int d = 0) {
    //cout << v << " " << p << " " << d << endl;
    cur += d;
    for(auto to : g[v]) {
        if(to != p)
            dfs(to, v, d + 1);
    }
}

void check() {
    for(int i = 0; i < n; i++) {
        g[i].clear();
    }
    for(int i = 0; i < n; i++) {
        for(auto j : son[i]) {
            g[i].push_back(j);
            g[j].push_back(i);
        }
    }
    for(int i = 0; i < n; i++) {
        sort(g[i].begin(), g[i].end());
    }
    int h = get_h(g);
    if(s.count(h))
        return;
    s.emplace(h);
    cur = 0;
    for(int i = 0; i < n; i++) {
        //for(auto to : g[i])
        //cout << i << " " << to << endl;
        dfs(i);
    }
    cur /= 2;
    //cout << cur << endl;
    kek[cur]++;
}

vector<int> p;
void rec(int v) {
    if(v == n) {
```

```

        check();
        return;
    }
    for (int i = 0; i < v; i++) {
        son[p[i]].push_back(p[v]);
        rec(v + 1);
        son[p[i]].pop_back();
    }
}

int main() {
#ifdef MY
    freopen("input.txt", "r", stdin);
    freopen("output.txt", "w", stdout);
#endif // MY
    ios_base::sync_with_stdio(false);
    cin.tie(0);
    cout << fixed << setprecision(5);
    s.max_load_factor(0.5);
    s.reserve(1000);
    double start = clock();
    cin >> n >> m;
    p.resize(n);
    g.resize(n);
    for (int i = 0; i < n; i++)
        p[i] = i;
    int fact = 1;
    for (int i = 2; i <= n; i++)
        fact *= i;
    for (int i = 0; i < fact; i++) {
        double cur = clock();
        if ((cur - start) / CLOCKS_PER_SEC > 2.9)
            break;
        random_shuffle(p.begin(), p.end());
        rec(1);
    }
    for (int i = 1; i <= m; i++) {
        cout << kek[i] << " ";
    }
    return 0;
}

```