

Олимпиада СПбГУ по информатике 2019/20 учебного года

A	B	C	D	E	F	Sum
100	100	100	40	100	0	440

Task A ()

```
n = int(input())  
print(n - 1)
```

Task B ()

```
from math import *

class Vector:
    def __init__(self, a, b):
        self.x = a
        self.y = b

    def __add__(self, other):
        return Vector(self.x + other.x, self.y + other.y)

    def rotate(self, angle):
        return Vector(self.x * cos(angle) - self.y * sin(angle),
                       self.y * cos(angle) + self.x * sin(angle))

    def dist2(self, other):
        return pow(self.x - other.x, 2) + pow(self.y - other.y, 2)

    def __mul__(self, k):
        return Vector(self.x * k, self.y * k)

n = int(input())
points = list()

for i in range(n):
    x, y = map(float, input().split())
    points.append(Vector(x, y))

d = 0.0
p1 = Vector(-1, -1)
p2 = Vector(-1, -1)

for i in range(n):
    for j in range(i + 1, n):
        d1 = points[i].dist2(points[j])
        if d1 < d:
            p1 = points[i]
            p2 = points[j]
            d = d1

p3 = p1 + Vector((p2.x - p1.x) / 2.0, (p2.y - p1.y) / 2.0)

if n == 6:
    for i in [p1, p2, p3]:
        print("%.10f_%.10f" % (i.x, i.y))

if n == 3:
    v = Vector(p3.x - p1.x, p3.y - p1.y)
    for i in range(6):
        p = p3 + v
        v = v.rotate(radians(60))
        print("%.10f_%.10f" % (p.x, p.y))
```

Task C ()

```
t = input()
n = int(input())
ans = 0
for _ in range(n):
    s = input()
    lans = len(t)
    for beg in range(len(s)):
        pos = beg
        llans = 0

        for i in range(len(t)):
            if pos < len(s) and s[pos] == t[i]:
                pos += 1
            else:
                llans += 1

        lans = min(lans, llans)

    ans += lans
print(ans)
```

Task D ()

```
#include <iostream>
#include <vector>
#include <math.h>
#include <algorithm>
#include <string>
#include <map>
#include <numeric>

#pragma warning(disable : 4996)
using namespace std;

typedef long long ll;

#define all(a) (a).begin(), (a).end()
#define mineq(a, b) (a) = min(a, b)

void IO() {
#ifdef SEVA
    freopen("input.txt", "r", stdin);
    freopen("output.txt", "w", stdout);
#endif
    ios_base::sync_with_stdio(false);
    cin.tie(NULL);
    cout.tie(NULL);
    cout.setf(ios::fixed);
    cout.precision(10);
}

struct Vector {
    int x, y;

    Vector(int a, int b) : x(a), y(b) {};
    Vector() : Vector(0, 0) {};

    Vector operator + (const Vector& a) const {
        return Vector(x + a.x, y + a.y);
    }

    bool operator != (const Vector& a) const {
        return x != a.x || y != a.y;
    }
};

const int N = 1010;

Vector field[N][N];

bool used[N][N];
int dist[N][N];

void solve() {
    int n, m;
    cin >> n >> m;
    Vector a, b;
    cin >> a.x >> a.y;
    cin >> b.x >> b.y;
    a.x--;
    a.y--;
    b.x--;
    b.y--;

    for (int i = 0; i < n; i++) {
        for (int j = 0; j < m; j++) {
            int x, y;
            cin >> x >> y;
            field[i][j] = { x, y };
        }
    }

    for (int i = 0; i < n; i++) {
        for (int j = 0; j < m; j++) {
```

```

        dist[i][j] += abs(b.x - i - field[i][j].x);
        dist[i][j] += abs(b.y - j - field[i][j].y);
    }
}
dist[b.x][b.y] = 0;

while (true) {
    int x = -1;
    int y = -1;
    for (int i = 0; i < n; i++) {
        for (int j = 0; j < m; j++) {
            if (used[i][j])
                continue;
            if (x == -1 || dist[i][j] < dist[x][y]) {
                x = i;
                y = j;
            }
        }
    }
    if (x == -1)
        break;
    used[x][y] = true;
    for (int i = 0; i < n; i++) {
        for (int j = 0; j < m; j++) {
            int lans = abs(x - i - field[i][j].x) + abs(y - j - field[i][j].y)
                ;
            mineq(dist[i][j], lans + dist[x][y]);
        }
    }
}
cout << dist[a.x][a.y] << endl;
}

int main() {
    IO();
    solve();
    return 0;
}

```

Task E ()

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

#pragma warning(disable : 4996)
using namespace std;

typedef long long ll;

void IO() {
#ifdef _SEVA
    freopen("input.txt", "r", stdin);
    freopen("output.txt", "w", stdout);
#endif
    ios_base::sync_with_stdio(false);
    cin.tie(NULL);
    cout.tie(NULL);
    cout.setf(ios::fixed);
    cout.precision(10);
}

struct Point {
    ll x, y;

    bool operator < (const Point& a) const {
        return x < a.x || (x == a.x && y < a.y);
    }
};

const int N = 1 << 13;

pair<Point, bool> fields[N];

void solve() {
    ll n, m, b;
    cin >> n >> m >> b;
    vector<Point> a(b);
    for (auto& i : a) {
        cin >> i.x >> i.y;
        i.x--;
        i.y--;
    }

    for (int i = 0; i < (1 << b); i++) {
        fields[i].second = true;
        fields[i].first = { i * 1LL * n, i * 1LL * m };
    }
    map<Point, int> ids;
    for (int i = b - 1; i >= 0; i--) {
        int cnt = (1 << i);
        int t = 0;
        Point p1, p2;
        for (int id = 0; id < (1 << b) && cnt > 0; id++) {
            if (!fields[id].second)
                continue;
            ll x = fields[id].first.x + a[i].x;
            ll y = fields[id].first.y + a[i].y;
            ids[{x, y}] = id;
            if (t == 0)
                p1 = { x, y };
            else
                p2 = { x, y };
            t++;
            if (t == 2) {
                t = 0;
                cnt--;
                cout << "?_ " << p1.x << "_ " << p1.y << "_ " << p2.x << "_ " << p2.y
                    << endl;
                ll xa, ya;
            }
        }
    }
}
```

```

        cin >> xa >> ya;
        fields[ids[{xa, ya}]].second = false;
    }
}
for (int id = 0; id < (1 << b); id++) {
    if (!fields[id].second)
        continue;
    cout << "!_" << fields[id].first.x << "_" << fields[id].first.y << endl;
    return;
}
}

int main() {
    //IO();
    solve();
    return 0;
}

```

Task F ()

```
#include <iostream>
#include <vector>
#include <math.h>
#include <algorithm>
#include <string>
#include <map>
#include <numeric>
#include <set>

#pragma warning(disable : 4996)
using namespace std;

typedef long long ll;

#define all(a) (a).begin(), (a).end()

void IO() {
#ifdef SEVA
    freopen("input.txt", "r", stdin);
    freopen("output.txt", "w", stdout);
#endif
    ios_base::sync_with_stdio(false);
    cin.tie(NULL);
    cout.tie(NULL);
    cout.setf(ios::fixed);
    cout.precision(10);
}

const int N = 26;
const int M = (int)1e9 + 7;
const int p = 1103;

int binpow(int a, int n) {
    int ans = 1;
    while (n) {
        if (n & 1)
            ans = (ll(ans) * a) % ll(M);
        a = (ll(a) * a) % ll(M);
        n >>= 1;
    }
    return ans;
}

void solve() {
    ll n, m;
    cin >> n >> m;
    vector<int> ans(m, 0);
    int fac = 1;
    for (int i = 1; i <= n; i++) {
        fac = (ll(fac) * i) % ll(M);
    }
    for (ll i = m - 1, t = 2; i >= 0; i--, t++) {
        fac = (ll(fac) * binpow(t, M - 2)) % ll(M);
        if (i + 1 >= (n - 1 + (n - 1) * (n - 2)))
            ans[i] = fac;
        else
            ans[i] = 0;
    }

    for (auto& i : ans) {
        cout << i << "_";
    }
}

int main() {
    IO();
    solve();
    return 0;
}

//446193002
```