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 sqrt


def sz(x):
    return sqrt(x[0] ** 2 + x[1] ** 2)


def get(x1, y1, x2, y2, x3, y3):
    v1 = (x2 - x1, y2 - y1)
    p1 = (-v1[1], v1[0])
    p1 = (p1[0] / sz(p1), p1[1] / sz(p1))
##    print(v1)
##    print(p1)
    mul = -(2 / sqrt(3)) * sz(v1)
    p1 = (p1[0] * mul, p1[1] * mul)
##    print(p1, mul)
    return (x3 + p1[0], y3 + p1[1])


n = int(input())
if n == 3:
    x1, y1 = map(float, input().split())
    x2, y2 = map(float, input().split())
    x3, y3 = map(float, input().split())
    d1 = get(x1, y1, x2, y2, x3, y3)
    d2 = get(x3, y3, x1, y1, x2, y2)
    d3 = get(x2, y2, x3, y3, x1, y1)
    print('%1.3f %1.3f' % (x1, y1))
    print('%1.3f %1.3f' % d1)
    print('%1.3f %1.3f' % (x2, y2))
    print('%1.3f %1.3f' % d3)
    print('%1.3f %1.3f' % (x3, y3))
    print('%1.3f %1.3f' % d2)
else:
    d = []
    for i in range(6):
        d.append(tuple(map(float, input().split())))
    r = []
    d.sort(key=lambda x: (-x[1], x[0]))
    c = d[0]
    ms = 100000000
    v = -1
    r.append(c)
    for i in range(1, 6):
        ms1 = sz((d[i][0] - c[0], d[i][1] - c[1]))
        if abs(ms - ms1) < 0.01:
            if d[v][0] > d[i][0]:
                v = i
        elif ms1 < ms:
            v = i
            ms = ms1
    c = d[v]
    for i in range(4):
        r.append(c)
        ms = 10000000
        v = -1
        for j in range(6):
            if d[j] not in r:
                ms1 = ms
                ms = min(ms, sz((d[j][0] - c[0], d[j][1] - c[1])))
                if ms < ms1:
                    v = j
        c = d[v]
    r.append(c)
    #r.reverse()
    for i in range(6):
        if i % 2 == 0:
            print(*r[i])
```

Task C ()

```
def ins(ind):
    cnt = 0
    i = ind
    for j in range(len(t)):
        if i < len(s) and s[i] == t[j]:
            i += 1
        else:
            cnt += 1
    return cnt


t = list(input())
n = int(input())
ans = 0
for _ in range(n):
    s = list(input())
    ans1 = len(t)
    for i in range(len(s)):
        ans1 = min(ans1, ins(i))
    ans += ans1
    #print(ans1)
print(ans)
```

Task D ()

```
#include <bits/stdc++.h>
using namespace std;

int main(){
    int n, m, sx, sy, fx, fy, INF = 1000000000, a, b;
    cin >> n >> m >> sx >> sy >> fx >> fy;
    sx -= 1;
    sy -= 1;
    fx -= 1;
    fy -= 1;
    vector< vector< pair<int, int> > > w(n);
    vector< vector < int > > d(n);
    for (int i = 0; i < n; ++i){
        for (int j = 0; j < m; ++j){
            d[i].push_back(INF);
            cin >> a >> b;
            w[i].push_back(make_pair(a, b));
        }
    }
    d[sx][sy] = 0;
    vector< pair<int, int> > q;
    int q0 = 0, x, y, vx, vy, s;
    q.push_back(make_pair(sx, sy));
    while (q.size() > q0){
        x = q[q0].first;
        y = q[q0].second;
        q0 += 1;
        for (int i = 0; i < n; ++i){
            for (int j = 0; j < m; ++j){
                vx = i - x - w[x][y].first;
                vy = j - y - w[x][y].second;
                s = abs(vx) + abs(vy);
                if (d[i][j] > s + d[x][y]){
                    d[i][j] = s + d[x][y];
                    q.push_back(make_pair(i, j));
                }
            }
        }
    }
    cout << d[fx][fy] << endl;
    /*for (int i = 0; i < n; ++i){
        for (int j = 0; j < m; ++j){
            cout << d[i][j] << " ";
        }
        cout << endl;
    }*/
}
```

Task E ()

```
def paint(ind):
    if c[ind] == -1:
        return -1
    j = c[ind]
    return (d[ind][0] + bl[j][0], d[ind][1] + bl[j][1])


n, m, b = map(int, input().split())
##if b == 13:
##    print(1 / 0)
d = [0] * (2 ** b)
c = [0] * (2 ** b)
for i in range(2 ** (b - b // 2)):
    for j in range(2 ** (b // 2)):
        d[(2 ** (b // 2)) * i + j] = (n * i, m * j)
        c[(2 ** (b // 2)) * i + j] = 0
bl = []
for i in range(b):
    bl.append(tuple(map(int, input().split())))
for i in range(b):
    p = 0
    for j in range(2 ** (b - i - 1)):
        while p < 2 ** b and paint(p) == -1:
            p += 1
        if p >= 2 ** b:
            break
        x1, y1 = paint(p)
        p1 = p
        p += 1
        while p < 2 ** b and paint(p) == -1:
            p += 1
        if p >= 2 ** b:
            break
        x2, y2 = paint(p)
        p2 = p
        p += 1
        print('?', x1, y1, x2, y2)
        c[p1] += 1
        c[p2] += 1
        v1, v2 = map(int, input().split())
        v1 -= 1
        v2 -= 1
        c[(v1 // n) * (2 ** (b // 2)) + v2 // m] = -1
for i in range(len(c)):
    if c[i] == b:
        print('!', d[i][0] + 1, d[i][1] + 1)
        break
```

Task F ()