

Олимпиада СПбГУ по информатике 2019/20 учебного года

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

Task A ()

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

using namespace std;

typedef long long ll;
typedef long double ld;
#define pb push_back
#define pll pair<ll, ll>

void solve() {
    ll n;
    cin >> n;
    cout << n - 1 << endl;
}

int main()
{
    //freopen("input.txt", "r", stdin);
    //freopen("output.txt", "w", stdout);

    ios::sync_with_stdio(0);
    cin.tie(0);
    cout.tie(0);

    solve();

    return 0;
}
```

Task B ()

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

using namespace std;

typedef long long ll;
typedef long double ld;
#define pb push_back
#define pll pair<ll, ll>

vector<pair<ld, ld>> a;

ld dist(ll i, ll j) {
    ld dx = a[i].first - a[j].first;
    ld dy = a[i].second - a[j].second;
    return sqrt(dx * dx + dy * dy);
}

void solve() {
    ll n;
    cin >> n;
    for(ll i = 0; i < n; i++) {
        ld x, y;
        cin >> x >> y;
        a.pb({x, y});
    }
    if(n == 6) {
        vector<pair<ld, ll>> cur;
        for(ll i = 1; i < 6; i++) {
            ld val = dist(0, i);
            cur.pb({val, i});
        }
        sort(cur.begin(), cur.end());
        vector<ll> ps = {cur[0].second, 0, cur[1].second};
        for(auto el : ps) {
            cout << fixed << setprecision(4) << a[el].first << " " << a[el].second << endl;
        }
    } else {
        ld dx = a[2].first - a[1].first;
        ld dy = a[2].second - a[1].second;

        a.pb({a[0].first + 2 * dx, a[0].second + 2 * dy});

        ld dx2 = a[1].first - a[0].first;
        ld dy2 = a[1].second - a[0].second;
        a.pb({a[3].first - dx2, a[3].second - dy2});

        a.pb({a[4].first - dx, a[4].second - dy});

        for(auto el : a) {
            cout << fixed << setprecision(4) << el.first << " " << el.second << endl;
        }
    }
}

int main()
{
    //freopen("input.txt", "r", stdin);
    //freopen("output.txt", "w", stdout);

    ios::sync_with_stdio(0);
    cin.tie(0);
    cout.tie(0);

    solve();

    return 0;
}
```

Task C ()

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

using namespace std;

typedef long long ll;
typedef long double ld;
#define pb push_back
#define pll pair<ll, ll>

void solve() {
    string t;
    cin >> t;
    ll n;
    cin >> n;
    ll ans = 0;
    for (ll i = 0; i < n; i++) {
        string s;
        cin >> s;
        ll res = t.size();
        for (ll j = 0; j < s.size(); j++) {
            ll cur = t.size();
            ll post = 0, poss = j;
            while (post < t.size() && poss < s.size()) {
                while (post < t.size() && t[post] != s[poss]) {
                    post++;
                }
                if (post == t.size()) break;
                post++;
                poss++;
                cur--;
            }
            res = min(res, cur);
        }
        ans += res;
    }
    cout << ans << endl;
}

int main()
{
    //freopen("input.txt", "r", stdin);
    //freopen("output.txt", "w", stdout);

    ios::sync_with_stdio(0);
    cin.tie(0);
    cout.tie(0);

    solve();

    return 0;
}
```

Task D ()

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

using namespace std;

typedef long long ll;
typedef long double ld;
#define pb push_back
#define pll pair<ll, ll>

const ll maxn = 1001;
const ll inf = 1e9;

ll n, m;
pll a[maxn][maxn];

void solve() {
    cin >> n >> m;
    ll fx, fy, tx, ty;
    cin >> fx >> fy >> tx >> ty;
    fx--;
    fy--;
    tx--;
    ty--;
    for (ll i = 0; i < n; i++) {
        for (ll j = 0; j < m; j++) {
            ll x, y;
            cin >> x >> y;
            a[i][j] = {x, y};
        }
    }

    ll dist[maxn][maxn];
    ll used[maxn][maxn];
    for (ll i = 0; i < n; i++) {
        for (ll j = 0; j < m; j++) {
            dist[i][j] = inf;
            used[i][j] = 0;
        }
    }
    dist[fx][fy] = 0;

    for (ll tt = 0; tt < n * m; tt++) {
        ll mn = inf;
        pll pos;
        for (ll i = 0; i < n; i++) {
            for (ll j = 0; j < m; j++) {
                if (used[i][j]) continue;
                if (dist[i][j] < mn) {
                    mn = dist[i][j];
                    pos = {i, j};
                }
            }
        }
        used[pos.first][pos.second] = 1;
        for (ll i = 0; i < n; i++) {
            for (ll j = 0; j < m; j++) {
                ll cur = abs(i - pos.first - a[pos.first][pos.second].first) +
                    abs(j - pos.second - a[pos.first][pos.second].second) +
                    dist[pos.first][pos.second];
                dist[i][j] = min(dist[i][j], cur);
            }
        }
    }

    cout << dist[tx][ty] << endl;
}

int main()
{
    //freopen("input.txt", "r", stdin);
    //freopen("output.txt", "w", stdout);
}
```

```
ios::sync_with_stdio(0);  
cin.tie(0);  
cout.tie(0);  
  
solve();  
  
return 0;  
}
```

Task E ()

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

using namespace std;

typedef long long ll;
typedef long double ld;
#define pb push_back
#define pll pair<ll, ll>

vector<pll> recs;
vector<ll> ban;
map<pll, ll> to;
vector<ll> used;

ll qs = 0;

void ask(ll x1, ll y1, ll x2, ll y2) {
    qs++;
    cout << "?_ " << x1 << "_ " << y1 << "_ " << x2 << "_ " << y2 << endl;
    ll x, y;
    cin >> x >> y;
    ban[to[{x, y}]] = 1;
}

pll find_firsts() {
    pll res = {-1, -1};
    for (ll i = 0; i < recs.size(); i++) {
        if (!ban[i] && !used[i]) {
            res.first = i;
            used[i] = true;
            break;
        }
    }
    for (ll i = res.first + 1; i < recs.size(); i++) {
        if (!ban[i] && !used[i]) {
            res.second = i;
            used[i] = true;
            break;
        }
    }
    return res;
}

void solve() {
    ll n, m, b;
    cin >> n >> m >> b;
    vector<pll> ps;
    for (ll i = 0; i < b; i++) {
        ll x, y;
        cin >> x >> y;
        x--;
        y--;
        ps.pb({x, y});
    }

    ll start = ll(0);
    ll skip = ll(3e5) + 5;

    ll cnt = 8192;
    for (ll i = 0; i < cnt; i += 2) {
        ll rx = start + skip * i;
        ll ry = start;
        recs.pb({rx, ry});
        ban.pb(0);
        recs.pb({rx + skip, ry});
        ban.pb(0);

        ll x = rx + ps[0].first;
        ll y = ry + ps[0].second;

        to[{x, y}] = i;
        to[{x + skip, y}] = i + 1;
    }
}
```

```

        ask(x, y, x + skip, y);
    }

    used.assign(cnt, 0);
    if(b == 1) {
        pll cur = find_firsts();
        cout << "!_" << recs[cur.first].first << "_" << recs[cur.first].second << endl;
        exit(0);
    }

    for(ll tt = 1; tt < b; tt++) {
        used.assign(cnt, 0);
        for(ll i = 0; i < cnt / (1 << (tt + 1)); i++) {
            pll cur = find_firsts();
            if(cur.second == -1) {
                if(cur.first != -1) {
                    ban[cur.first] = 1;
                }
                break;
            }

            ll x1 = recs[cur.first].first + ps[tt].first;
            ll y1 = recs[cur.first].second + ps[tt].second;
            to[{x1, y1}] = cur.first;

            ll x2 = recs[cur.second].first + ps[tt].first;
            ll y2 = recs[cur.second].second + ps[tt].second;
            to[{x2, y2}] = cur.second;

            ask(x1, y1, x2, y2);

            if(tt == b - 1) {
                if(ban[cur.first]) {
                    ll tmp = cur.second;
                    cur.second = cur.first;
                    cur.first = tmp;
                }
                cout << "!_" << recs[cur.first].first << "_" << recs[cur.first].second << endl;
                exit(0);
            }
        }
        for(ll i = 0; i < cnt; i++) {
            if(!used[i]) {
                ban[i] = true;
            }
        }
    }
}

int main()
{
    //freopen("input.txt", "r", stdin);
    //freopen("output.txt", "w", stdout);

    ios::sync_with_stdio(0);
    cin.tie(0);
    cout.tie(0);

    solve();

    return 0;
}

```

Task F ()