

# Олимпиада СПбГУ по информатике 2019/20 учебного года

A	B	C	D	E	F	Sum
100	100	100	20	0	0	320

## Task A ()

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

#pragma GCC optimize("O3")

#ifdef FAIRLY_LOCAL
#define io freopen("in.txt", "r", stdin); freopen("out.txt", "w", stdout)
#else
#define io ios_base::sync_with_stdio(0); cin.tie(0); cout.tie(0)
#endif // FAIRLY_LOCAL

#define forn(i, n) for (int i = 0; i < n; i++)
#define all(a) a.begin(), a.end()

typedef long long ll;
typedef long double ld;
// #define int ll

/*
vector<bool> used;
void bfs(int v) {
    used[v] = true;

    deque<int> q;
    q.push_back(v);

    while (!q.empty()) {
        v = q.front();
        q.pop_front();

        for (auto to : g[v]) {
            if (!used[to]) {
                used[to] = true;
                q.push_back(to);
            }
        }
    }
}
*/

signed main() {
    io;
    // cout << setprecision(23);

    ll n; cin >> n;
    if (n == 5) {
        cout << 4;
        return 0;
    }
    cout << n - 1;
}
```

## Task B ()

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

#pragma GCC optimize("O3")

#ifdef FAIRLY_LOCAL
#define io freopen("in.txt", "r", stdin); freopen("out.txt", "w", stdout)
#else
#define io ios_base::sync_with_stdio(0); cin.tie(0); cout.tie(0)
#endif // FAIRLY_LOCAL

#define forn(i, n) for (int i = 0; i < n; i++)
#define all(a) a.begin(), a.end()

typedef long long ll;
typedef long double ld;
// #define int ll

/*
vector<bool> used;
void bfs(int v) {
    used[v] = true;

    deque<int> q;
    q.push_back(v);

    while (!q.empty()) {
        v = q.front();
        q.pop_front();

        for (auto to : g[v]) {
            if (!used[to]) {
                used[to] = true;
                q.push_back(to);
            }
        }
    }
}
*/

#define pt pair<ld, ld>
#define x first
#define y second

/*
pt ctr;
bool cmp(pt a, pt b) {
    pt a1 = {a.x-ctr.x, a.y-ctr.y};
    pt b1 = {b.x-ctr.x, b.y-ctr.y};

    ld v1 = a1.x*
    return a.x * b.y - a.y * b.x < 0;
}
*/

ld sqr(ld a) {
    return a * a;
}

ld distsq(pt a, pt b) {
    return sqr(a.x - b.x) + sqr(a.y - b.y);
}

pt p;
bool cmp(pt a, pt b) {
    return distsq(a, p) < distsq(b, p);
}

signed main() {
    io;
```

```

cout << setprecision(23);

int n; cin >> n;
vector<pt> a(n);

if(n == 6) {
    forn(i, n) cin >> a[i].x >> a[i].y;
    p = a[0];
    sort(all(a), cmp);
    cout << setprecision(23) << a[0].x << '\u' << setprecision(23) << a[0].y << '\n';
    cout << setprecision(23) << (a[0].x+a[5].x) / 2 << '\u' << setprecision(23) << (a[0].y + a
        [5].y) / 2 << '\n';
    cout << setprecision(23) << a[5].x << '\u' << setprecision(23) << a[5].y << '\n';
}
else {
    forn(i, n) cin >> a[i].x >> a[i].y;

    //ctr = {a[1].x - 1e4, a[1].y - 1e4};

    pt perp = {-a[2].y+a[0].y, a[2].x - a[0].x};
    ld del = sqrt(sqr(perp.x) + sqr(perp.y));
    perp.x /= del, perp.y /= del;

    ld len = sqrt(sqr(a[2].x - a[0].x) + sqr(a[2].y - a[0].y)) / 2;
    //cout << "len " << len << '\n';
    perp.x*= sqrt(3)*len / 2, perp.y*= sqrt(3)*len / 2;
    //cout << "perp " << perp.x << ' ' << perp.y << '\n';

    pt diag = {(a[2].x - a[0].x) / 4, (a[2].y - a[0].y) / 4};
    //cout << "diag " << diag.x << ' ' << diag.y << '\n';

    a.emplace_back(a[1].x - diag.x + perp.x, a[1].y - diag.y + perp.y);
    a.emplace_back(a[1].x + diag.x + perp.x, a[1].y + diag.y + perp.y);
    a.emplace_back(a[1].x + diag.x - perp.x, a[1].y + diag.y - perp.y);
    a.emplace_back(a[1].x - diag.x - perp.x, a[1].y - diag.y - perp.y);

    a.erase(a.begin()+1);
    a.emplace(a.begin() + 4, a[1]);
    a.erase(a.begin()+1);
    //swap(a[1], a[3]);
    //3 10 0 0 0 -10 -0

    //sort(a.begin(), a.end(), cmp);
    forn(i, a.size()) cout << setprecision(23) << a[i].x << '\u' << setprecision(23) << a[i].y
        << '\n';
}
}

```

## Task C ()

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

#pragma GCC optimize("O3")

#ifdef FAIRLY_LOCAL
#define io freopen("in.txt", "r", stdin); freopen("out.txt", "w", stdout)
#else
#define io ios_base::sync_with_stdio(0); cin.tie(0); cout.tie(0)
#endif // FAIRLY_LOCAL

#define forn(i, n) for (int i = 0; i < n; i++)
#define all(a) a.begin(), a.end()

typedef long long ll;
typedef long double ld;
// #define int ll

/*
vector<bool> used;
void bfs(int v) {
    used[v] = true;

    deque<int> q;
    q.push_back(v);

    while (!q.empty()) {
        v = q.front();
        q.pop_front();

        for (auto to : g[v]) {
            if (!used[to]) {
                used[to] = true;
                q.push_back(to);
            }
        }
    }
}
*/

signed main() {
    io;
    // cout << setprecision(23);
    // prank 1 kaban

    string s; cin >> s;
    int q; cin >> q;

    ll ans = 0;
    string t;
    forn(z, q) {
        cin >> t;
        ll ls = 0, currAns = 1000, ctr = 0;

        for (int i = 0; i < t.size(); i++) {
            int lt = i; // ls = 0, ctr = 0;
            while (lt < t.size()) {
                while (s[ls] != t[lt] && ls < s.size() - 1) {
                    // cout << lt << t[lt] << ' ' << ls << s[ls] << '\n';
                    ls++;
                }
                // cout << lt << t[lt] << ' ' << ls << s[ls] << '\n';
                if (ls >= s.size() - 1 || (lt == t.size() - 1 && t[lt] == s[ls])) {
                    if (s[ls] == t[lt]) ctr++;
                    // cout << "exiting now, " << int(s.size()) - ctr << "\n\n";
                    currAns = min(currAns, int(s.size()) - ctr);
                    ls = 0, ctr = 0;
                    break;
                }
            }
            ctr++, lt++, ls++;
        }
    }
}
```

```
        }  
        ans+= currAns;  
    }  
    cout << ans;  
}
```

## Task D ()

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

#pragma GCC optimize("O3")

#ifdef FAIRLY_LOCAL
#define io freopen("in.txt", "r", stdin); freopen("out.txt", "w", stdout)
#else
#define io ios_base::sync_with_stdio(0); cin.tie(0); cout.tie(0)
#endif // FAIRLY_LOCAL

#define forn(i, n) for (int i = 0; i < n; i++)
#define all(a) a.begin(), a.end()

typedef long long ll;
typedef long double ld;
// #define int ll

#define pt pair<int, int>
#define x first
#define y second

int n, m;
pt s, f;
vector<vector<pt>>> g;
vector<vector<ll>>> dist;

void drive(pt p) {
    /*
    forn(i, n) {
        forn(j, m) {
            cout<< dist[i][j] << ' ';
        }
        cout << '\n';
    }
    cout << "\n\n";

    */
    forn(i, n) {
        forn(j, m) {
            dist[i][j] = min(dist[i][j], dist[p.x][p.y] + abs(i - p.x - g[p.x][p.y].x) + abs(j - p.y - g[p.x][p.y].y));
        }
    }
}

signed main() {
    io;

    cout << setprecision(23);

    cin >> n >> m >> s.x >> s.y >> f.x >> f.y;
    s.x--, s.y--, f.x--, f.y--;

    g.resize(n, vector<pt>(m));
    dist.assign(n, vector<ll>(m, 1e5));
    dist[s.x][s.y] = 0;

    forn(i, n) {
        forn(j, m) {
            cin >> g[i][j].x >> g[i][j].y;
        }
    }

    drive(s);
    forn(i, 10) {
```

```

    forn(i, n) {
        forn(j, m) {
            drive({i, j});
            // cout << "in ";
        }
    }
}

/*
forn(i, n) {
    forn(j, m) {
        cout << dist[i][j] << ' ';
    }
    cout << '\n';
}*/
cout << dist[f.x][f.y];
}

```

## Task E ()



## Task F ()