

Олимпиада СПбГУ по информатике 2019/20 учебного года

A	B	C	D	E	F	Sum
100	100	100	0	100	0	400

Task A ()

```
#include <iostream>
#include <vector>
#include <algorithm>
#include <queue>

#define ll long long
#define ld long double
#define io cin.sync_with_stdio(false);cout.sync_with_stdio(false);cin.tie(0);cout.tie(0);
#define point pair<ld,ld>
#define x first
#define y second

using namespace std;

signed main() {
    io;
    ll n;
    cin >> n;
    cout << n - 1;
    //system("PAUSE");
    return 0;
}
```

Task B ()

```
#include <iostream>
#include <vector>
#include <algorithm>
#include <queue>

#define ll long long
#define ld long double
#define io cin.sync_with_stdio(false);cout.sync_with_stdio(false);cin.tie(0);cout.tie(0);
#define point pair<ld,ld>
#define x first
#define y second

using namespace std;

ld sqd(point a, point b) {
    return (a.x - b.x) * (a.x - b.x) + (a.y - b.y) * (a.y - b.y);
}

signed main() {
    io;
    int n;
    cin >> n;
    cout.precision(3);
    if (n == 6) {
        vector<point> a(6);
        point cent;
        for (int i = 0; i < 6; i++) {
            cin >> a[i].x >> a[i].y;
            cent.x += a[i].x;
            cent.y += a[i].y;
        }
        cent.x /= (ld)6;
        cent.y /= (ld)6;
        cout << fixed << cent.x << "\u" << cent.y << "\n";
        cout << fixed << a[0].x << "\u" << a[0].y << "\n";
        for (auto cur : a) {
            if (abs(sqd(cent, a[0]) - sqd(a[0], cur)) < (ld)1) {
                cout << fixed << cur.x << "\u" << cur.y << "\n";
                //system("PAUSE");
                return 0;
            }
        }
        //system("PAUSE");
        return 1;
    } else {
        point a, b, c;
        cin >> a.x >> a.y >> b.x >> b.y >> c.x >> c.y;
        cout << fixed << b.x << "\u" << b.y << "\n";
        cout << fixed << c.x << "\u" << c.y << "\n";
        cout << fixed << a.x + c.x - b.x << "\u" << a.y + c.y - b.y << "\n";
        cout << fixed << 2 * a.x - b.x << "\u" << 2 * a.y - b.y << "\n";
        cout << fixed << 2 * a.x - c.x << "\u" << 2 * a.y - c.y << "\n";
        cout << fixed << a.x - c.x + b.x << "\u" << a.y - c.y + b.y << "\n";
    }
    //system("PAUSE");
    return 0;
}
```

Task C ()

```
#include <iostream>
#include <vector>
#include <algorithm>
#include <queue>
#include <string>

#define ll long long
#define ld long double
#define io cin.sync_with_stdio(false);cout.sync_with_stdio(false);cin.tie(0);cout.tie(0);
#define point pair<ld,ld>
#define f first
#define s second
#define all(a) a.begin(), a.end()

using namespace std;

signed main() {
    io;
    string t;
    cin >> t;
    int n;
    cin >> n;
    vector<string> a(n);
    for (int i = 0; i < n; i++)
        cin >> a[i];
    vector<vector<int>>> inn(256);
    for (int i = 0; i < t.size(); i++) {
        inn[t[i]].push_back(i);
    }
    int ans = 0;
    for (auto cur : a){
        int cnt = 0;
        vector<vector<pair<int, int>>>> pre(cur.size());
        if (!inn[cur[0]].empty()) {
            pre[0].emplace_back(inn[cur[0]][0], 1);
            cnt = 1;
        }
        for (int i = 1; i < cur.size(); i++) {
            if (inn[cur[i]].empty())
                continue;
            //if (pre[i-1].empty() || pre[i-1][0].f > inn[cur[i]][0]) {
            pre[i].emplace_back(inn[cur[i]][0], 1);
            cnt = max(cnt, 1);
            //}
            for (auto cc : pre[i-1]) {
                auto ii = upper_bound(all(inn[cur[i]]), cc.f);
                if (ii != inn[cur[i]].end()) {
                    pre[i].emplace_back(*ii, cc.s + 1);
                    cnt = max(cnt, cc.s + 1);
                }
            }
        }
        ans += t.size() - cnt;
    }
    cout << ans;
    //system("PAUSE");
    return 0;
}
```

Task D ()

```
#include <iostream>
#include <vector>
#include <algorithm>
#include <queue>
#include <string>
#include <set>

#define int long long
#define ld long double
#define io cin.sync_with_stdio(false);cout.sync_with_stdio(false);cin.tie(0);cout.tie(0);
#define point pair<int,int>
#define x first
#define y second
#define all(a) a.begin(), a.end()

using namespace std;

signed main() {
    io;
    int n, m;
    cin >> n >> m;
    vector<int> a(m);
    int ax, ay, bx, by;
    cin >> ax >> ay >> bx >> by;
    for (int i = 0; i < m; i++) {
        int x, y;
        cin >> x >> y;
        if (x == 0)
            a[i] = y;
    }
    int ans = by - ay;
    for (int i = ay - 1; i < by; i++) {
        if (ay < by) {
            if (a[i] == 1) {
                ans--;
            }
        }
        else {
            if (a[i] == -1) {
                ans++;
            }
        }
    }
    cout << abs(ans);
    return 0;
}
```

Task E ()

```
#include <iostream>
#include <vector>
#include <algorithm>
#include <queue>
#include <string>
#include <set>

#define int long long
#define ld long double
#define io cin.sync_with_stdio(false);cout.sync_with_stdio(false);cin.tie(0);cout.tie(0);
#define point pair<int,int>
#define x first
#define y second
#define all(a) a.begin(), a.end()

using namespace std;

int n, m, b, nee;

signed main() {
    io;
    cin >> n >> m >> b;
    vector<point> a(b);
    for (int i = 0; i < b; i++) {
        cin >> a[i].y >> a[i].x;
        a[i].x--;
        a[i].y--;
    }
    nee = ((1ll) << b);
    set<pair<int,int>> s;
    for (int i = 0; i < nee; i++) {
        s.insert({ 0, i });
    }
    vector<int> ans(nee);
    int ccc = 0;
    while ((*s.begin()).x != b) {
        ccc++;
        auto cur = (*s.begin());
        s.erase(cur);
        if ((*s.begin()).x == b)
            continue;
        cout << "?_";
        cout << a[cur.x].y + n * cur.y << "_ " << a[cur.x].x << "_";
        cur.x++;
        ans[cur.y] = cur.x;
        s.insert(cur);
        cur = (*s.begin());
        if ((*s.begin()).x == b) {
            cout << "-1_-1" << endl;
        }
        else {
            s.erase(cur);
            cout << a[cur.x].y + n * cur.y << "_ " << a[cur.x].x << endl;
            cur.x++;
            ans[cur.y] = cur.x;
            s.insert(cur);
        }
        int x, y;
        cin >> y >> x;
        if (y < 0) {
            s.erase(*s.begin());
            continue;
        }
        int num = y / n;
        if (num >= nee) {
            s.erase(*s.begin());
            continue;
        }
        if (s.find({ ans[num], num }) != s.end())
            s.erase({ ans[num], num });
        else
            s.erase(*s.begin());
    }
```

```
}  
cout << "!\n" << n * (*s.begin()).y << "\n0" << endl;  
//system("PAUSE");  
return 0;  
}
```

Task F ()