

Олимпиада СПбГУ по информатике 2022/23 учебного года

A	B	C	D	E	F	Sum
100	100	60	60	28	10	358

Task A ()

```
///#pragma GCC optimize("Ofast")
///#pragma GCC target("sse,sse2,sse3,ssse3,sse4.1,sse4.2,avx,avx2")

#include <bits/stdc++.h>

using namespace std;
mt19937_64 rnd(time(0));
using ll = long long;
const ll inf = 1e9;
#define all(x) (x).begin(), (x).end()

void solve() {
    vector<ll> b(6);
    for (auto &i : b) {
        cin >> i, i--;
    }

    vector<ll> a = {1, 2, 3, 4, 5, 6};
    do {
        vector<pair<ll, ll>> now;
        bool f = 1;
        for (int i = 0; i < 6; i++) {
            now.push_back({a[i], i});
            sort(all(now));
            ll needed_pos = b[i];
            if (now[needed_pos].second == i)
                continue;
            else {
                f = 0;
                break;
            }
        }
        if (f) {
            for (auto &i : a) {
                cout << i << ' ';
            }
            return;
        }
    } while (next_permutation(all(a)));
}

int main() {
    cin.tie(0);
    cout.tie(0);
    ios_base::sync_with_stdio(0);
    ll tt = 1;
    //cin >> tt;
    while (tt--) {
        solve();
    }
    return 0;
}
```

Task B ()

```
//#pragma GCC optimize("Ofast")
//#pragma GCC target("sse,sse2,sse3,ssse3,sse4.1,sse4.2,avx,avx2")

#include <bits/stdc++.h>

using namespace std;
mt19937_64 rnd(time(0));
using ll = long long;
const ll inf = 1e9;
#define all(x) (x).begin(), (x).end()

void solve_first() {
    ll n;
    cin >> n;
    vector<ll> a(n);
    ll sum = 0;
    for (auto &i : a) {
        cin >> i;
        sum += i;
    }
    ll DEL = 1e9;
    cout << sum / n + (sum % n) * DEL;
}

void solve_second() {
    ll n;
    cin >> n;
    vector<ll> a(n);
    ll DEL = 1e9, res = 0;
    for (auto &i : a) {
        cin >> i;
    }
    ll k = a[0] / DEL;
    for (auto &i : a) {
        ll before = i - k * DEL;
        res += before;
    }
    res += k;
    cout << res;
}

void solve() {
    string type;
    cin >> type;
    if (type == "first") {
        solve_first();
    } else {
        solve_second();
    }
}

int main() {
    cin.tie(0);
    cout.tie(0);
    ios_base::sync_with_stdio(0);
    ll tt = 1;
    //cin >> tt;
    while (tt--) {
        solve();
    }
    return 0;
}
```

Task C ()

```
//#pragma GCC optimize("Ofast")
//#pragma GCC target("sse,sse2,sse3,ssse3,sse4.1,sse4.2,avx,avx2")

#include <bits/stdc++.h>

using namespace std;
mt19937_64 rnd(time(0));
using ll = long long;
const ll inf = 1e9;
#define all(x) (x).begin(), (x).end()
ll n = 3;
vector<ll> di = {-1, 0, 1, 0};
vector<ll> dj = {0, 1, 0, -1};
vector<pair<int, int>> a(n);
set<pair<ll, ll>> can;
vector<vector<ll>> used;

void add(int x, int y) {
    can.insert({min(x, y), max(x, y)});
}

struct node {
    int par;
    int min_i, min_j, max_i, max_j;
    int cnt;
    node() : par(-1), min_i(100), min_j(100), max_i(-100), max_j(-100), cnt(0) {}
    node(int par, int min_i, int min_j, int max_i, int max_j, int cnt)
        : par(par), min_i(min_i), min_j(min_j), max_i(max_i), max_j(max_j), cnt(cnt) {}
};

vector<node> sys;

struct item {
    int i, j, a, b;
    item() : i(-1), j(-1), a(-1), b(-1) {}
    item(int i, int j, int a, int b) : i(i), j(j), a(a), b(b) {}
};

bool check_item(item f) {
    bool left_right = 0;
    bool up_down = 0;
    for (int i = f.i; i <= f.i + f.a - 1; i++) {
        if (used[i][f.j - 1] || used[i][(f.j + f.b - 1) + 1])
            left_right |= 1;
    }
    for (int j = f.j; j <= f.j + f.b - 1; j++) {
        if (used[f.i - 1][j] || used[(f.i + f.a - 1) + 1][j])
            up_down |= 1;
    }
    return (left_right + up_down) >= 2;
}

ll num(ll i, ll j) {
    return (i - 1) * a[0].second + j;
}

int find(int v) {
    if (sys[v].par == v)
        return v;
    return sys[v].par = find(sys[v].par);
}

void unite(int v, int u) {
    v = find(v), u = find(u);
    if (u == v)
        return;
    sys[v].cnt += sys[u].cnt;
    sys[v].min_i = min(sys[v].min_i, sys[u].min_i);
    sys[v].min_j = min(sys[v].min_j, sys[u].min_j);
    sys[v].max_i = max(sys[v].max_i, sys[u].max_i);
    sys[v].max_j = max(sys[v].max_j, sys[u].max_j);
}
```

```

    sys[u].par = v;
}

void create(int v, int i, int j) {
    sys[v].par = v;
    sys[v].min_i = sys[v].max_i = i;
    sys[v].min_j = sys[v].max_j = j;
    sys[v].cnt = 1;
}

void make(item f, item s) {
    used.clear();
    sys.clear();
    used.resize(a[0].first + 3, vector<ll>(a[0].second + 3, 1));
    sys.resize((a[0].first + 3) * (a[0].second + 3) + 1);
    for (int i = 1; i <= a[0].first; i++) {
        for (int j = 1; j <= a[0].second; j++) {
            used[i][j] = 0;
            ll c = num(i, j);
            create(c, i, j);
        }
    }

    if (f.i >= 1 && f.j >= 1) {
        if (f.i + f.a - 1 > a[0].first || f.j + f.b - 1 > a[0].second)
            return;
        if (!check_item(f))
            return;
        for (int i = f.i; i <= f.i + f.a - 1; i++) {
            for (int j = f.j; j <= f.j + f.b - 1; j++) {
                used[i][j]++;
                if (used[i][j] >= 2)
                    return;
            }
        }
    }

    if (s.i >= 1 && s.j >= 1) {
        if (s.i + s.a - 1 > a[0].first || s.j + s.b - 1 > a[0].second)
            return;
        if (!check_item(s))
            return;
        for (int i = s.i; i <= s.i + s.a - 1; i++) {
            for (int j = s.j; j <= s.j + s.b - 1; j++) {
                used[i][j]++;
                if (used[i][j] >= 2)
                    return;
            }
        }
    }

    for (int i = 1; i <= a[0].first; i++) {
        for (int j = 1; j <= a[0].second; j++) {
            if (used[i][j])
                continue;
            for (int d = 0; d < 4; d++) {
                int i1 = i + di[d], j1 = j + dj[d];
                if (i1 >= 1 && j1 >= 1 && i1 <= a[0].first && j1 <= a[0].second && !used[i1][j1])
                    unite(num(i, j), num(i1, j1));
            }
        }
    }

    for (int i = 1; i <= a[0].first; i++) {
        for (int j = 1; j <= a[0].second; j++) {
            if (used[i][j])
                continue;
            ll v = find(num(i, j));
            ll A = sys[v].max_i - sys[v].min_i + 1, b = sys[v].max_j - sys[v].min_j + 1;
            if (A * b == sys[v].cnt) {
                add(A, b);
            }
        }
    }
}

```

```

    }
}

void solve() {
    for (auto &i : a) {
        cin >> i.first >> i.second;
        add(i.first, i.second);
    }

    do {
        for (int i = 1; i <= a[0].first; i++) {
            for (int j = 1; j <= a[0].second; j++) {
                item f1 = {i, j, a[1].first, a[1].second};
                item f2 = {i, j, a[1].second, a[1].first};
                item fake = item();
                make(f1, fake), make(f2, fake);
                for (int i1 = 1; i1 <= a[0].first; i1++) {
                    for (int j1 = 1; j1 <= a[0].second; j1++) {
                        item s1 = {i1, j1, a[2].first, a[2].second};
                        item s2 = {i1, j1, a[2].second, a[2].first};
                        make(f1, s1), make(f1, s2), make(f2, s1), make(f2, s2);
                    }
                }
            }
        }
        swap(a[0].first, a[0].second);
        for (int i = 1; i <= a[0].first; i++) {
            for (int j = 1; j <= a[0].second; j++) {
                item f1 = {i, j, a[1].first, a[1].second};
                item f2 = {i, j, a[1].second, a[1].first};
                item fake = item();
                make(f1, fake), make(f2, fake);
                for (int i1 = 1; i1 <= a[0].first; i1++) {
                    for (int j1 = 1; j1 <= a[0].second; j1++) {
                        item s1 = {i1, j1, a[2].first, a[2].second};
                        item s2 = {i1, j1, a[2].second, a[2].first};
                        make(f1, s1), make(f1, s2), make(f2, s1), make(f2, s2);
                    }
                }
            }
        }
        swap(a[0].first, a[0].second);
    } while (next_permutation(all(a)));
    do {
        for (int i = 1; i <= a[0].first; i++) {
            for (int j = 1; j <= a[0].second; j++) {
                item f1 = {i, j, a[1].first, a[1].second};
                item f2 = {i, j, a[1].second, a[1].first};
                item fake = item();
                make(f1, fake), make(f2, fake);
                for (int i1 = 1; i1 <= a[0].first; i1++) {
                    for (int j1 = 1; j1 <= a[0].second; j1++) {
                        item s1 = {i1, j1, a[2].first, a[2].second};
                        item s2 = {i1, j1, a[2].second, a[2].first};
                        make(f1, s1), make(f1, s2), make(f2, s1), make(f2, s2);
                    }
                }
            }
        }
        swap(a[0].first, a[0].second);
        for (int i = 1; i <= a[0].first; i++) {
            for (int j = 1; j <= a[0].second; j++) {
                item f1 = {i, j, a[1].first, a[1].second};
                item f2 = {i, j, a[1].second, a[1].first};
                item fake = item();
                make(f1, fake), make(f2, fake);
                for (int i1 = 1; i1 <= a[0].first; i1++) {
                    for (int j1 = 1; j1 <= a[0].second; j1++) {
                        item s1 = {i1, j1, a[2].first, a[2].second};
                        item s2 = {i1, j1, a[2].second, a[2].first};
                        make(f1, s1), make(f1, s2), make(f2, s1), make(f2, s2);
                    }
                }
            }
        }
        swap(a[0].first, a[0].second);
    }
}

```

```

        }
        swap(a[0].first , a[0].second);
    } while (next_permutation(all(a)));

    for(auto &i : can) {
        cout << i.first << ' ' << i.second << '\n';
    }
}

int main() {
    cin.tie(0);
    cout.tie(0);
    ios_base::sync_with_stdio(0);
    ll tt = 1;
    //cin >> tt;
    while (tt--) {
        solve();
    }
    return 0;
}

```

Task D ()

```
//#pragma GCC optimize("Ofast")
//#pragma GCC target("sse,sse2,sse3,ssse3,sse4.1,sse4.2,avx,avx2")

#include <bits/stdc++.h>

using namespace std;
mt19937_64 rnd(time(0));
using ll = long long;
const ll N = 1e4 + 5;
#define all(x) (x).begin(), (x).end()
#define sz(x) (int)(x).size()

void solve() {
    ll n;
    cin >> n;
    vector<ll> a(n);
    for (auto &i : a) {
        cin >> i;
    }
    if (n == 1) {
        cout << 1 << ' ' << a[0] << endl;
        ll p, val;
        cin >> p >> val;
        cout << 1 << ' ' << a[0] << endl;
        cin >> p >> val;
        return;
    }
    vector<ll> times(2, 1);
    while (true) {
        if ((a[0] + a[1]) || (times[0] + times[1])) {
            if ((a[0] ^ a[1]) == 0) {
                if (times[0]) {
                    cout << 1 << ' ' << 0 << endl;
                } else if (times[1]) {
                    cout << 2 << ' ' << 0 << endl;
                } else {
                    if (a[0])
                        cout << 1 << ' ' << 1 << endl;
                    else
                        cout << 2 << ' ' << 1 << endl;
                }
                ll p, val;
                cin >> p >> val;
                if (p == -1) {
                    return;
                }
                p--;
                a[p] -= val;
            }
        }
        cout << "-1_-1" << endl;
        return;
    }
}

int main() {
    cin.tie(0);
    cout.tie(0);
    ios_base::sync_with_stdio(0);
    ll tt = 1;
    string type;
    //cin >> tt;
    while (tt--) {
        solve();
    }
    return 0;
}
```

Task E ()

```
//#pragma GCC optimize("Ofast")
//#pragma GCC target("sse,sse2,sse3,ssse3,sse4.1,sse4.2,avx,avx2")

#include <bits/stdc++.h>

using namespace std;
mt19937_64 rnd(time(0));
using ll = long long;
const ll N = 1e4 + 5;
#define all(x) (x).begin(), (x).end()
#define sz(x) (int)(x).size()
vector<ll> now(10, 0), info(10, 0);
ll n, lst = 0;

bool go_first(int level, int bef) {
    if (level == 10) {
        lst++;
        if (lst == n) {
            for (int i = 0; i < 10; i++) {
                for (int j = 0; j < now[i]; j++) {
                    cout << '1';
                }
                for (int j = now[i]; j < 10; j++) {
                    cout << '0';
                }
                cout << '\n';
            }
            return true;
        }
        return false;
    }
    for (int digit = bef; digit <= 10; digit++) {
        now[level] = digit;
        if (go_first(level + 1, digit))
            return true;
    }
    return false;
}

void solve_first() {
    cin >> n;
    lst = 0;
    go_first(0, 0);
    cout << '\n';
}

bool go_second(int level, int bef) {
    if (level == 10) {
        lst++;
        if (now == info) {
            cout << lst;
            return true;
        }
        return false;
    }
    for (int digit = bef; digit <= 10; digit++) {
        now[level] = digit;
        if (go_second(level + 1, digit))
            return true;
    }
    return false;
}

void solve_second() {
    for (int i = 0; i < 10; i++) {
        int cnt = 0;
        for (int j = 0; j < 10; j++) {
            char c;
            cin >> c;
            if (c == '1')
                cnt++;
        }
    }
}
```



```

        info[i] = cnt;
    }
    sort(all(info));
    lst = 0;
    go_second(0, 0);
    cout << '\n';
}

int main() {
    cin.tie(0);
    cout.tie(0);
    ios_base::sync_with_stdio(0);
    ll tt = 1;
    string type;
    cin >> tt >> type;
    while (tt--) {
        if (type == "transmit") {
            solve_first();
        } else {
            solve_second();
        }
    }
    return 0;
}

```

Task F ()

```
//#pragma GCC optimize("Ofast")
//#pragma GCC target("sse,sse2,sse3,ssse3,sse4.1,sse4.2,avx,avx2")

#include <bits/stdc++.h>

using namespace std;
mt19937_64 rnd(time(0));
using ll = long long;
const ll inf = 1e18;
const ll N = 1e3 + 5;
#define all(x) (x).begin(), (x).end()
#define sz(x) (int)(x).size()

string get(string s) {
    string res = "";
    ll n = sz(s);
    for (int i = 0; i < n; i++) {
        if ('0' <= s[i] && s[i] <= '9') {
            res += s[i];
            i++;
            continue;
        }
        string klolik = "";
        i++;
        while ('0' <= s[i] && s[i] <= '9') {
            klolik += s[i];
            i++;
        }
        i++;
        ll times = 0;
        while ('0' <= s[i] && s[i] <= '9') {
            times *= 10;
            times += (s[i] - '0');
            i++;
        }
        i++;
        while (times-- > 0) {
            res += klolik;
        }
    }
    return res;
}

ll count(string s) {
    ll n = sz(s);
    ll sizee = 0;
    for (int i = 0; i < n; i++) {
        if ('0' <= s[i] && s[i] <= '9') {
            sizee++;
            i++;
            continue;
        }
        string klolik = "";
        i++;
        while ('0' <= s[i] && s[i] <= '9') {
            klolik += s[i];
            i++;
        }
        i++;
        ll times = 0;
        while ('0' <= s[i] && s[i] <= '9') {
            times *= 10;
            times += (s[i] - '0');
            i++;
        }
        sizee += times * sz(klolik);
        i++;
    }
    return sizee;
}

struct BigNum {
    ll t[N];
};
```

```

BigNum() {
    memset(t, 0, sizeof t);
}
BigNum(string s) {
    memset(t, 0, sizeof t);
    reverse(all(s));
    for (int i = 0; i < sz(s); i++) {
        t[i] = s[i] - '0';
    }
}
};

string make(BigNum a, BigNum b) {
    BigNum res = BigNum();

    for (int i = 0; i < N; i++) {
        res.t[i] = a.t[i] + b.t[i];
    }

    ll add = 0;

    for (int i = 0; i < N; i++) {
        res.t[i] += add;
        add = res.t[i] / 10;
        res.t[i] %= 10;
    }
    string klolik = "";
    for (int i = 0; i < N; i++) {
        klolik += char(res.t[i] + '0');
    }
    while (sz(klolik) > 1 && klolik.back() == '0')
        klolik.pop_back();
    reverse(all(klolik));
    return klolik;
}

struct item {
    ll l, r, d;
};

struct event {
    ll pos, digit, type, type_op;
    event() = default;
    event(ll pos, ll digit, ll type, ll type_op) : pos(pos), digit(digit), type(type), type_op(
        type_op) {}
    bool operator<(event &n) {
        if (pos == n.pos)
            return type_op < n.type_op;
        return pos < n.pos;
    }
};

vector<item> e;

vector<item> remake(string s) {
    reverse(all(s));
    vector<item> res;
    ll pos = 0;
    for (int i = 0; i < sz(s);) {
        if ('0' <= s[i] && s[i] <= '9') {
            res.push_back({pos, pos, s[i] - '0'});
            pos++;
            i++;
            continue;
        }
        i++;
        string times = "";
        while ('0' <= s[i] && s[i] <= '9') {
            times += s[i];
            i++;
        }
        ll TIMES = 0;
        for (int j = sz(times) - 1; j >= 0; j--) {
            TIMES *= 10;
            TIMES += (times[j] - '0');
        }
    }
}

```

```

    }
    i++;
    int digit = s[i] - '0';
    i++;
    i++;
    res.push_back({pos, pos + TIMES - 1, digit});
    pos += TIMES;
}
return res;
}

void solve() {
    string a, b;
    cin >> a >> b;
    if (count(a) <= 100 && count(b) <= 100) {
        BigNum A(get(a)), B(get(b));
        cout << make(A, B);
        return;
    }
    vector<item> for_a = remake(a);
    vector<item> for_b = remake(b);

    ll sumA = 0, sumB = 0, C;
    for (auto &i : for_a) {
        sumA += i.r - i.l + 1;
    }
    for (auto &i : for_b) {
        sumB += i.r - i.l + 1;
    }
    C = max(sumA, sumB) + 5;
    for_a.push_back({for_a.back().r + 1, C - 1, 0});
    for_b.push_back({for_b.back().r + 1, C - 1, 0});

    vector<event> e;
    for (auto &i : for_a) {
        e.emplace_back(i.l, i.d, 0, +1);
        e.emplace_back(i.r + 1, i.d, 0, -1);
    }
    for (auto &i : for_b) {
        e.emplace_back(i.l, i.d, 1, +1);
        e.emplace_back(i.r + 1, i.d, 0, -1);
    }
    sort(all(e));
    int d0 = -1, d1 = -1, carry = 0;
    vector<pair<ll, ll>> res;
    for (int i = 0; i < sz(e); i++) {
        if (e[i].type_op == +1) {
            if (e[i].type == 0)
                d0 = e[i].digit;
            else
                d1 = e[i].digit;
            continue;
        }
        if (carry == 0) {
            ll times = e[i].pos - e[i - 1].pos;
            if (times == 0)
                continue;
            if (d0 + d1 <= 9) {
                res.emplace_back(d0 + d1, times);
            } else {
                res.emplace_back((d0 + d1) % 10, 1);
                if (times - 1)
                    res.emplace_back((d0 + d1) % 10 + 1, times - 1);
                carry = 1;
            }
        } else {
            ll times = e[i].pos - e[i - 1].pos;
            if (times == 0)
                continue;
            if (d0 + d1 + 1 <= 9) {
                res.emplace_back(d0 + d1 + 1, 1);
                if (times - 1)
                    res.emplace_back(d0 + d1, times - 1);
                carry = 0;
            }
        }
    }
}

```

```

        }
        else {
            res.emplace_back((d0 + d1) % 10 + 1, times);
            carry = 1;
        }
    }
    reverse(all(res));

    for(auto &i : res) {
        cout << "(" << i.first << "|" << i.second << ")";
    }
}

int main() {
    cin.tie(0);
    cout.tie(0);
    ios_base::sync_with_stdio(0);
    ll tt = 1;
    //cin >> tt;
    while (tt--) {
        solve();
    }
    return 0;
}

```