

# Олимпиада СПбГУ по информатике 2022/23 учебного года

A	B	C	D	E	F	Sum
100	100	100	60	16	10	386

## Task A ()

```
/* Includes */
#include <bits/stdc++.h>
#include <ext/pb_ds/assoc_container.hpp>
#include <ext/pb_ds/tree_policy.hpp>

/* Using libraries */
using namespace std;
using namespace __gnu_pbds;

/* Some libraries utils */
template <class T>
using ordered_set = tree<T, null_type, less<T>, rb_tree_tag, tree_order_statistics_node_update>;

/* Defines */
#define fast ios_base::sync_with_stdio(false); cin.tie(0); cout.tie(0)
#define ld long long
#define vc vector
#define pb push_back
#define out(a) for (int i: a) cout << i << ' ';
#define graph(n, g) for (int i = 0; i < n; ++i) {int u, v; cin >> u >> v; --u; --v; g[u].push_back(v); g[v].push_back(u);}
#define eb emplace_back
#define pii pair<int, int>
#define all(a) a.begin(), a.end()
#define sz(a) (int) a.size()
#define forn(i, n) for(int i = 0; i < n; ++i)
#define int long long

const int inf = 1e13;
template <class T>
bool chmax (T &a, T b) {
    if (a < b) {
        a = b;
        return true;
    }
    return false;
}
template <class T>
bool chmin (T &a, T b) {
    if (a > b) {
        a = b;
        return true;
    }
    return false;
}

// 23fin0043
// plavionly

void solve () {
    int n = 6;
    vc<int> pos;
    for (int i = 0; i < n; ++i) {
        int j;
        cin >> j;
        if (j == sz(pos) + 1) {
```

```

        pos.pb(i);
        continue;
    }
    for (int k = 0; k < sz(pos); ++k) {
        if (k + 1 == j) {
            pos.insert(pos.begin() + k, i);
            break;
        }
    }
}
vc <int> ip(n);
for (int i = 0; i < n; ++i) {
    ip[pos[i]] = i;
}
for (int i : ip) {
    cout << i + 1 << '␣';
}
cout << '\n';
}

/* Starting and precalcing */
signed main() {
    // freopen("input.txt","r",stdin); /* freopen("output.txt","w",stdout); */
    fast;
    int t = 1;
    // cin >> t;
    while (t--) {
        solve();
    }
    return 0;
}

```

## Task B ()

```
/* Includes */
#include <bits/stdc++.h>
#include <ext/pb_ds/assoc_container.hpp>
#include <ext/pb_ds/tree_policy.hpp>

/* Using libraries */
using namespace std;
using namespace __gnu_pbds;

/* Some libraries utils */
template <class T>
using ordered_set = tree<T, null_type, less<T>, rb_tree_tag, tree_order_statistics_node_update>;

/* Defines */
#define fast ios_base::sync_with_stdio(false); cin.tie(0); cout.tie(0)
#define ld long long
#define vc vector
#define pb push_back
#define out(a) for (int i: a) cout << i << ' ';
#define graph(n, g) for (int i = 0; i < n; ++i) {int u, v; cin >> u >> v; --u; --v; g[u].push_back(v); g[v].push_back(u);}
#define eb emplace_back
#define pii pair<int, int>
#define all(a) a.begin(), a.end()
#define sz(a) (int) a.size()
#define forn(i, n) for(int i = 0; i < n; ++i)
#define int long long

const int inf = 1e13;
template <class T>
bool chmax (T &a, T b) {
    if (a < b) {
        a = b;
        return true;
    }
    return false;
}
template <class T>
bool chmin (T &a, T b) {
    if (a > b) {
        a = b;
        return true;
    }
    return false;
}

// 23fin0043
// plavionly

void solve () {
    string s;
    cin >> s;
    int n;
    cin >> n;
    int sum1 = 0, sum2 = 0;
    for (int i = 0; i < n; ++i) {
        int x;
        cin >> x;
        sum1 += x;
        sum2 += x % 1000;
    }
    if (s == "first") {
        int ret = 0;
        cout << sum1 * 1000 << '\n';
        return;
    }
    cout << (sum1 - sum2) / 1000 / n + sum2 << '\n';
}

/*
first
3
*/
```

```

1 2 4

second
3
3 4 6
*/

/* Starting and precalcing */
signed main() {
    // freopen("input.txt","r",stdin); /* freopen("output.txt","w",stdout); */
    fast;
    int t = 1;
    // cin >> t;
    while (t--) {
        solve();
    }
    return 0;
}

```

## Task C ()

```
/* Includes */
#include <bits/stdc++.h>
#include <ext/pb_ds/assoc_container.hpp>
#include <ext/pb_ds/tree_policy.hpp>

/* Using libraries */
using namespace std;
using namespace __gnu_pbds;

/* Some libraries utils */
template <class T>
using ordered_set = tree<T, null_type, less<T>, rb_tree_tag, tree_order_statistics_node_update>;

/* Defines */
#define fast ios_base::sync_with_stdio(false); cin.tie(0); cout.tie(0)
#define ld long long
#define vc vector
#define pb push_back
#define out(a) for (int i: a) cout << i << ' ';
#define graph(n, g) for (int i = 0; i < n; ++i) {int u, v; cin >> u >> v; --u; --v; g[u].push_back(v); g[v].push_back(u);}
#define eb emplace_back
#define pii pair<int, int>
#define all(a) a.begin(), a.end()
#define sz(a) (int) a.size()
#define forn(i, n) for(int i = 0; i < n; ++i)
#define int long long

const int inf = 1e13;
template <class T>
bool chmax (T &a, T b) {
    if (a < b) {
        a = b;
        return true;
    }
    return false;
}
template <class T>
bool chmin (T &a, T b) {
    if (a > b) {
        a = b;
        return true;
    }
    return false;
}

// 23fin0043
// plavionly

void solve () {
    int n = 3;
    vc<pii> v(n);
    for (auto &[x, y] : v) {
        cin >> x >> y;
        if (x > y)
            swap(x, y);
    }
    sort(all(v));
    vc<pii> ans = v;
    do {
        int n1 = v.back().first, m1 = v.back().second;
        --n;
        auto go = [&]() {
            auto [x1, y1] = v[0];
            auto [x2, y2] = v[1];
            if (x1 == n1 && y1 < m1) {
                ans.pb({n1, m1 - y1});
            }
            if (x1 * y1 + x2 * y2 >= n1 * m1)
                return;
            if (y1 + y2 > n1 && y1 < n1 && y2 < n1 && x1 <= n1 && x1 + x2 <= m1) {
                ans.pb({x1, n1 - y1});
            }
        };
    } while (n > 0);
}
```

```

    }
    if (x1 == x2 && y1 + y2 == n1 && x1 < m1) {
        ans.pb({n1, m1 - x1});
    }
    if (y1 == m1 && x1 + x2 == n1 && y2 < m1) {
        ans.pb({n1 - x1, m1 - y2});
    }
    if (x1 == n1 && x2 == n1 && y1 + y2 < m1) {
        ans.pb({n1, m1 - y1 - y2});
    }
    if (x1 + x2 == n1 && y1 + y2 == m1) {
        ans.pb({x1, y2});
        ans.pb({x2, y1});
    }
    if (ans.back() == make_pair(211, 211)) {
        cerr << n1 << ' ' << m1 << '\n';
        cerr << x1 << ' ' << y1 << '\n';
        cerr << x2 << ' ' << y2 << '\n';
        exit(0);
    }
};
for (int u = 0; u < 2; ++u) {
    swap(n1, m1);
    for (int k = 0; k < 2; ++k) {
        swap(v[0].first, v[0].second);
        for (int i = 0; i < 2; ++i) {
            swap(v[1].first, v[1].second);
            go();
        }
    }
} while (next_permutation(all(v)));
for (auto &[x, y] : ans) {
    if (x > y)
        swap(x, y);
}
sort(all(ans));

ans.resize(unique(all(ans)) - ans.begin());
for (auto [x, y] : ans)
    cout << x << ' ' << y << '\n';
}

/* Starting and precalculing */
signed main() {
    // freopen("input.txt", "r", stdin); // freopen("output.txt", "w", stdout);
    fast;
    int t = 1;
    // cin >> t;
    while (t--) {
        solve();
    }
    return 0;
}

```

## Task D ()

```

/* Includes */
#include <bits/stdc++.h>
#include <ext/pb_ds/assoc_container.hpp>
#include <ext/pb_ds/tree_policy.hpp>

/* Using libraries */
using namespace std;
using namespace __gnu_pbds;

/* Some libraries utils */
template <class T>
using ordered_set = tree<T, null_type, less<T>, rb_tree_tag, tree_order_statistics_node_update>;

/* Defines */
#define fast ios_base::sync_with_stdio(false); cin.tie(0); cout.tie(0)
#define ld long long
#define vc vector
#define pb push_back
#define out(a) for (int i: a) cout << i << ' ';
#define graph(n, g) for (int i = 0; i < n; ++i) {int u, v; cin >> u >> v; --u; --v; g[u].push_back(v); g[v].push_back(u);}
#define eb emplace_back
#define pii pair<int, int>
#define all(a) a.begin(), a.end()
#define sz(a) (int) a.size()
#define forn(i, n) for(int i = 0; i < n; ++i)

const int inf = 1e13;
template <class T>
bool chmax (T &a, T b) {
    if (a < b) {
        a = b;
        return true;
    }
    return false;
}
template <class T>
bool chmin (T &a, T b) {
    if (a > b) {
        a = b;
        return true;
    }
    return false;
}

struct game {
    int mask = 0;
    vc<int> a, b;
    game () {}
    bool operator < (game const o) const {
        if (mask == o.mask)
            return a < o.a;
        return mask < o.mask;
    }
};

map<game, bool> mp;
map<game, pii> hod;

bool go (game x) {
    if (mp.find(x) != mp.end())
        return mp[x];
    int n = sz(x.a);
    if (x.mask == (1 << n) - 1 && accumulate(all(x.a), 0ll) == 0)
        return mp[x] = false;
    mp[x] = false;
    int xr = 0;
    for (int i = 0; i < n; ++i) {
        xr ^= x.a[i];
    }
    for (int i = 0; i < n; ++i) {
        vc<int> nd;

```

```

        if ((xr ^ x.a[i]) < x.a[i])
            nd.pb(x.a[i] - (xr ^ x.a[i]));
    for (int j = 1; j <= x.a[i]; ++j) {
        nd.pb(j);
        if (j != x.a[i])
            nd.pb(x.a[i] - j);
    }
    sort(all(nd)); nd.resize(unique(all(nd)) - nd.begin());
    for (int j : nd) {
        game nw = x;
        nw.a[i] -= j;
        bool win = go(nw);
        if (!win) {
            mp[x] = true;
            hod[x] = {i, j};
            return true;
        }
    }
}
}
for (int i = 0; i < n; ++i) {
    if (!(x.mask >> i & 1)) {
        game nw = x;
        nw.mask = x.mask | (1 << i);
        nw.a[i] = x.b[i];
        bool win = go(nw);
        if (!win) {
            mp[x] = true;
            hod[x] = {i, 0};
            return true;
        }
    }
}
return mp[x];
}

void solve () {
    int n;
    cin >> n;
    vc<int> a(n), used(n);
    int xr = 0;
    for (int i = 0; i < n; ++i) {
        cin >> a[i];
        xr ^= a[i];
    }
    vc<int> b = a;
    game cur;
    cur.a = a;
    cur.b = b;
    cur.mask = 0;
    auto hd = [&](int i, int x){
        if (x == 0) {
            cur.a[i] = cur.b[i];
            cur.mask |= 1 << i;
        } else {
            cur.a[i] -= x;
        }
    };
    auto ju = [&]() {
        int i, x;
        cin >> i >> x;
        if (i == -1)
            exit(0);
        hd(i - 1, x);
    };
    auto cnt = [&]() {
        xr = 0;
        for (int i : a)
            xr ^= i;
    };
    if (xr == 0 && n % 2 == 0) {
        cout << "-1_-1" << endl;
        return;
    }
    if (n == 3) {

```



```

    if (xr == 0) {
        cout << "1_0" << endl;
        hd(0, 0);
        ju();
    } else {
        for (int i = 0; i < 0; ++i) {
            int nd = a[i] ^ xr;
            if (nd < a[i]) {
                cout << i + 1 << ' ' << a[i] - nd << endl;
                hd(i, a[i] - nd);
                ju();
                break;
            }
        }
    }
}

bool win = go(cur);
if (!win) {
    cout << "-1_-1" << endl;
    return;
}
while (1) {
    if (!mp[cur]) {
        cout << "-1_-1" << endl;
        return;
    }
    pii h = hod[cur];
    cout << h.first + 1 << ' ' << h.second << endl;
    hd(h.first, h.second);
    ju();
}
}

/* Starting and precalculing */
signed main() {
    // freopen("input.txt", "r", stdin); // freopen("output.txt", "w", stdout);
    fast;
    int t = 1;
    // cin >> t;
    while (t--) {
        solve();
    }
    return 0;
}

```

## Task E ()

```

/* Includes */
#include <bits/stdc++.h>
#include <ext/pb_ds/assoc_container.hpp>
#include <ext/pb_ds/tree_policy.hpp>

/* Using libraries */
using namespace std;
using namespace __gnu_pbds;

/* Some libraries utils */
template <class T>
using ordered_set = tree<T, null_type, less<T>, rb_tree_tag, tree_order_statistics_node_update>;

/* Defines */
#define fast ios_base::sync_with_stdio(false); cin.tie(0); cout.tie(0)
#define ld long long
#define vc vector
#define pb push_back
#define out(a) for (int i: a) cout << i << ' ';
#define graph(n, g) for (int i = 0; i < n; ++i) {int u, v; cin >> u >> v; --u; --v; g[u].push_back(v); g[v].push_back(u);}
#define eb emplace_back
#define pii pair<int, int>
#define all(a) a.begin(), a.end()
#define sz(a) (int) a.size()
#define forn(i, n) for(int i = 0; i < n; ++i)
#define int long long

const int inf = 1e13;
template <class T>
bool chmax (T &a, T b) {
    if (a < b) {
        a = b;
        return true;
    }
    return false;
}
template <class T>
bool chmin (T &a, T b) {
    if (a > b) {
        a = b;
        return true;
    }
    return false;
}

// 23fin0043
// plavionly

string who;

int sum_x (int x, int b) {
    int s = 0;
    while (x > 0) {
        s += x % b;
        x /= b;
    }
    return s;
}

void solve () {
    if (who[0] == 't') {
        int x;
        cin >> x;
        vc <vc<int>> a(10, vc<int>(10));
        vc<int> e;
        int b;
        for (int i = 0; i < 10; ++i) {
            e.pb(x >> i & 1);
        }
        for (int i = 1; i < sz(e); ++i) {
            e[i] += e[i - 1];
        }
    }
}

```

```

    }
    for (int i = 0; i < sz(e); ++i) {
        for (int j = 0; j < e[i]; ++j) {
            a[i][j] = 1;
        }
    }
    for (int i = 0; i < 10; ++i) {
        for (int j = 0; j < 10; ++j)
            cout << a[i][j];
        cout << '\n';
    }
    cout << '\n';
} else {
    int ret = 0;
    vc <int> sum1(10), sum2(10);
    for (int i = 0; i < 10; ++i) {
        string s; cin >> s;
        for (int j = 0; j < 10; ++j) {
            sum1[i] += s[j] - '0';
            sum2[j] += s[j] - '0';
        }
    }
    sort(all(sum1));
    for (int i = 0; i < 10; ++i) {
        int y = sum1[i] - (i == 0 ? 0 : sum1[i - 1]);
        if (y == 1)
            ret += 1 << i;
    }
    cout << ret << '\n';
}
}

/*
first
3
1 2 4

second
3
3 4 6
*/

/* Starting and precalculing */
signed main() {
    // freopen("input.txt", "r", stdin); /* freopen("output.txt", "w", stdout); */
    fast;
    int t = 1;
    cin >> t >> who;
    while (t--) {
        solve();
    }
    return 0;
}

```

## Task F ()

```
/* Includes */
#include <bits/stdc++.h>
#include <ext/pb_ds/assoc_container.hpp>
#include <ext/pb_ds/tree_policy.hpp>

/* Using libraries */
using namespace std;
using namespace __gnu_pbds;

/* Some libraries utils */
template <class T>
using ordered_set = tree<T, null_type, less<T>, rb_tree_tag, tree_order_statistics_node_update>;

/* Defines */
#define fast ios_base::sync_with_stdio(false); cin.tie(0); cout.tie(0)
#define ld long long
#define vc vector
#define pb push_back
#define out(a) for (int i: a) cout << i << ' ';
#define graph(n, g) for (int i = 0; i < n; ++i) {int u, v; cin >> u >> v; --u; --v; g[u].push_back(v); g[v].push_back(u);}
#define eb emplace_back
#define pii pair<int, int>
#define all(a) a.begin(), a.end()
#define sz(a) (int) a.size()
#define forn(i, n) for(int i = 0; i < n; ++i)
#define int long long

const int inf = 1e13;
template <class T>
bool chmax (T &a, T b) {
    if (a < b) {
        a = b;
        return true;
    }
    return false;
}
template <class T>
bool chmin (T &a, T b) {
    if (a > b) {
        a = b;
        return true;
    }
    return false;
}

vc <pii> get_segs (string s) {
    int n = sz(s);
    vc <pii> segs;
    for (int i = 0; i < n; ++i) {
        if (s[i] == '(') {
            string x = ""; ++i;
            while (s[i] != '|') {
                x += s[i++];
            }
            ++i;
            string y = "";
            while (s[i] != ')') {
                y += s[i++];
            }
            segs.pb({stoll(x), stoll(y)});
        } else {
            segs.pb({s[i] - '0', 1});
        }
    }
    reverse(all(segs));
    return segs;
}

string unhash (vc <pii> segs) {
    reverse(all(segs));
    string x;
```

```

    for (int i = 0; i < sz(segs); ++i) {
        if (segs[i].second == 1) {
            x += to_string(segs[i].first);
        } else {
            x += "(" + to_string(segs[i].first) + "|" + to_string(segs[i].second) + ";";
        }
    }
    reverse(all(x));
    while (x.back() == '0')
        x.pop_back();
    reverse(all(x));
    return x;
}

void solve () {
    string a, b;
    cin >> a >> b;
    vc<pii> segs1 = get_segs(a);
    vc<pii> segs2 = get_segs(b);
    int len1 = 0, len2 = 0;
    for (auto [x, y] : segs1) {
        len1 += y;
    }
    for (auto [x, y] : segs2) {
        len2 += y;
    }
    if (len1 < len2)
        segs1.pb({0, len2 - len1});
    if (len2 < len1)
        segs2.pb({0, len1 - len2});
    vc<pii> res_segs;
    {
        int i = 0, j = 0;
        int add = 0;
        int cur1 = 0, cur2 = 0;
        while (i < sz(segs1) && j < sz(segs2)) {
            int l = min(cur1, cur2);
            int r = min(cur1 + segs1[i].second, cur2 + segs2[j].second);
            int d1 = segs1[i].first, d2 = segs2[j].first;
            if (d1 + d2 + add >= 10) {
                if (add) {
                    res_segs.pb({d1 + d2 + add - 10, r - 1});
                } else {
                    res_segs.pb({d1 + d2 - 10, 1});
                    if (r - l > 1) {
                        res_segs.pb({d1 + d2 + 1 - 10, r - l - 1});
                    }
                }
                add = 1;
            } else {
                if (add) {
                    res_segs.pb({d1 + d2 + 1, 1});
                    if (r - l > 1) {
                        res_segs.pb({d1 + d2, r - l - 1});
                    }
                } else {
                    res_segs.pb({d1 + d2, r - l});
                }
                add = 0;
            }
            if (cur1 + segs1[i].second == r) {
                ++i;
            } else {
                segs1[i].second = cur1 + segs1[i].second - r;
            }
            if (cur2 + segs2[j].second == r) {
                ++j;
            } else {
                segs2[j].second = cur2 + segs2[j].second - r;
            }
            cur1 = cur2 = r;
        }
        if (add)
            res_segs.pb({1, 1});
    }
}

```

```

    }
    cout << unhash(res_segs) << '\n';
}

/* Starting and precalcing */
signed main() {
    // freopen("input.txt","r",stdin); /* freopen("output.txt","w",stdout); */
    fast;
    int t = 1;
    // cin >> t;
    while (t--) {
        solve();
    }
    return 0;
}

```