

Олимпиада СПбГУ по информатике 2022/23 учебного года

| A | B | C | D | E | F | Sum |
|-----|-----|-----|----|----|----|-----|
| 100 | 100 | 100 | 10 | 52 | 10 | 372 |

Task A ()

```
#include "bits/stdc++.h"
/**pragma GCC optimize("Ofast,no-stack-protector")
#pragma GCC target("sse,sse2,sse3,ssse3,sse4,popcnt,abm,mmx,avx,avx2")
#pragma GCC optimize("unroll-loops")*/
using ll = long long;
using ld = long double;
using namespace std;
#define ff first
#define ss second
#define sz(a) (int)(a).size()
#define forn(i, n) for (int i = 0; i < (int)n; ++i)
#define pb push_back
#define all(a) begin(a), end(a)
#ifdef LOCAL
#define debug(a) cerr << #a << ":\n" << a << endl
#else
#define debug(a)
#endif

mt19937 rnd(chrono::steady_clock::now().time_since_epoch().count());
const int INFi = 1e9 + 228;
const ll INFll = (ll)1e18 + 228;

template <typename T>
bool inmin(T& a, T b) {
    if (a <= b) return false;
    a = b; return true;
}

template <typename T>
bool inmax(T& a, T b) {
    if (a >= b) return false;
    a = b; return true;
}

void solve();

signed main() {
    ios_base::sync_with_stdio(0), cin.tie(0);
    cout << fixed << setprecision(10);
#ifdef LOCAL
    freopen("/Users/alexsus/Desktop/solve/read.txt", "r", stdin);
#endif
    int tc;
    tc = 1;
    //cin >> tc;
    forn(i, tc) {
        solve();
    }
}

void solve() {
    vector<int> a(6);
    forn(i, 6) cin >> a[i];
    set<int> ord;
    forn(i, 6) ord.insert(i);
```

```

vector<int> ans(6);
int last = 0;
while (!ord.empty()) {
    last++;
    int val = INF;
    int pos;
    for (int x : ord) {
        if (val >= a[x]) {
            val = a[x];
            pos = x;
        }
    }
    for (int i = pos + 1; i < 6; ++i) {
        a[i]--;
    }
    ord.erase(pos);
    ans[pos] = last;
}
for (int i = 0; i < 6; ++i) cout << ans[i] << ' ';
}

```

Task B ()

```
#include "bits/stdc++.h"
/**pragma GCC optimize("Ofast,no-stack-protector")
#pragma GCC target("sse,sse2,sse3,ssse3,sse4,popcnt,abm,mmx,avx,avx2")
#pragma GCC optimize("unroll-loops")*/
using ll = long long;
using ld = long double;
using namespace std;
#define ff first
#define ss second
#define sz(a) (int)(a).size()
#define forn(i, n) for (int i = 0; i < (int)n; ++i)
#define pb push_back
#define all(a) begin(a), end(a)
#ifdef LOCAL
#define debug(a) cerr << #a << ":\n" << a << endl
#else
#define debug(a)
#endif

mt19937 rnd(chrono::steady_clock::now().time_since_epoch().count());
const int INFi = 1e9 + 228;
const ll INFll = (ll)1e18 + 228;

template <typename T>
bool inmin(T& a, T b) {
    if (a <= b) return false;
    a = b; return true;
}

template <typename T>
bool inmax(T& a, T b) {
    if (a >= b) return false;
    a = b; return true;
}

void solve();

signed main() {
    ios_base::sync_with_stdio(0), cin.tie(0);
    cout << fixed << setprecision(10);
#ifdef LOCAL
    freopen("/Users/alexsus/Desktop/solve/read.txt", "r", stdin);
#endif
    int tc;
    tc = 1;
    //cin >> tc;
    forn(i, tc) {
        solve();
    }
}

int base = 100 * 100 + 1;

void solve() {
    string w;
    cin >> w;
    if (w == "first") {
        int n;
        cin >> n;
        ll sm = 0;
        forn(i, n) {
            ll x;
            cin >> x;
            sm += x;
        }
        ll g = sm % n;
        ll av = sm / n;
        ll num = av * base * base + g * base;
        cout << num << endl;
    } else {
```

```

    int n;
    cin >> n;
    ll sm = 0;
    vector<ll> b(n);
    forn(i, n) {
        cin >> b[i];
        sm += b[i];
    }
    vector<ll> vals;
    while (sm > 0) {
        vals.pb(sm % base);
        sm /= base;
    }
    reverse(all(vals));
    ll was = vals[0];
    was += vals[1] / n;
    was += vals[2];
    cout << was << endl;
}
}

```

Task C ()

```
#include "bits/stdc++.h"
/**pragma GCC optimize("Ofast,no-stack-protector")
#pragma GCC target("sse,sse2,sse3,ssse3,sse4,popcnt,abm,mmx,avx,avx2")
#pragma GCC optimize("unroll-loops")*/
using ll = long long;
using ld = long double;
using namespace std;
#define ff first
#define ss second
#define sz(a) (int)(a).size()
#define forn(i, n) for (int i = 0; i < (int)n; ++i)
#define pb push_back
#define all(a) begin(a), end(a)
#ifdef LOCAL
#define debug(a) cerr << #a << ":\n" << a << endl
#else
#define debug(a)
#endif

mt19937 rnd(chrono::steady_clock::now().time_since_epoch().count());
const int INFi = 1e9 + 228;
const ll INFll = (ll)1e18 + 228;

template <typename T>
bool inmin(T& a, T b) {
    if (a <= b) return false;
    a = b; return true;
}

template <typename T>
bool inmax(T& a, T b) {
    if (a >= b) return false;
    a = b; return true;
}

void solve();

signed main() {
    ios_base::sync_with_stdio(0), cin.tie(0);
    cout << fixed << setprecision(10);
#ifdef LOCAL
    freopen("/Users/alexsus/Desktop/solve/read.txt", "r", stdin);
#endif
    int tc;
    tc = 1;
    //cin >> tc;
    forn(i, tc) {
        solve();
    }
}

#define rec pair<int, int>
bool used[10][10];
int dx[4] = {0, 0, -1, 1};
int dy[4] = {1, -1, 0, 0};
int n, m;

void dfs(int x, int y, int &cnt, vector<vector<int>> &a) {
    cnt++;
    used[x][y] = true;
    forn(k, 4) {
        int nx = x + dx[k];
        int ny = y + dy[k];
        if (nx >= 0 && nx < n && ny >= 0 && ny < m) {
            if (!used[nx][ny] && a[nx][ny] == 0) dfs(nx, ny, cnt, a);
        }
    }
}

void check(vector<vector<int>> &a, vector<rec> &ans) {
    n = sz(a), m = sz(a[0]);
    vector<rec> st;
    vector<rec> dv;
```

```

st.pb({0, 0});
dv.pb({1, 1});
st.pb({n - 1, 0});
dv.pb({-1, 1});
st.pb({0, m - 1});
dv.pb({1, -1});
st.pb({n - 1, m - 1});
dv.pb({-1, -1});
for(n(i, n) forn(j, m) used[i][j] = 0;
forn(i, 4) {
    auto [x, y] = st[i];
    if (used[x][y] || a[x][y]) continue;
    int now = 0;
    int cnta = 0, cntb = 0;
    int nx = x, ny = y;
    while (nx >= 0 && nx < n && !a[nx][y]) {
        cnta++;
        nx += dv[i].ff;
    }
    while (ny >= 0 && ny < m && !a[x][ny]) {
        cntb++;
        ny += dv[i].ss;
    }
    dfs(x, y, now, a);
    if (now == (cnta * cntb)) {
        ans.pb({cnta, cntb});
    }
}
}

void make(vector<rec> &b, vector<int> &path, vector<rec> &ans) {
    if (sz(path) == 0) return;
    if (sz(path) == 1) {
        ans.pb(b[0]);
        return;
    }
    vector<rec> st(sz(path));
    st[0] = {0, 0};
    vector<vector<int>> cnt(b[0].ff, vector<int> (b[0].ss));

    auto in = [&](int id) {
        return (st[id].ff + b[id].ff) <= b[0].ff && (st[id].ss + b[id].ss) <= b[0].ss;
    };

    for (int x = 0; x < b[0].ff; ++x) {
        for (int y = 0; y < b[0].ss; ++y) {
            st[1] = {x, y};
            bool ok1 = x == 0 || ((x + b[1].ff - 1) == (b[0].ff - 1));
            bool ok2 = y == 0 || ((y + b[1].ss - 1) == (b[0].ss - 1));
            if (in(1) && ok1 && ok2) {
                for (int i = x; i < x + b[1].ff; ++i) {
                    for (int j = y; j < y + b[1].ss; ++j) {
                        cnt[i][j] = 1;
                    }
                }
                if (sz(path) == 2) {
                    check(cnt, ans);
                } else {
                    for (int x2 = 0; x2 < b[0].ff; ++x2) {
                        for (int y2 = 0; y2 < b[0].ss; ++y2) {
                            st[2] = {x2, y2};
                            if (!in(2)) continue;
                            bool ok0 = true;
                            for (int ii = x2; ii < x2 + b[2].ff; ++ii) {
                                for (int jj = y2; jj < y2 + b[2].ss; ++jj) {
                                    ok0 &= cnt[ii][jj] == 0;
                                }
                            }
                            if (!ok0) continue;
                            bool ok1 = false;
                            bool ok2 = false;
                            for (int i = x2; i < x2 + b[2].ff; ++i) {
                                int d = y2;
                                int u = y2 + b[2].ss - 1;

```


} }

Task D ()

```
#include "bits/stdc++.h"
/**pragma GCC optimize("Ofast,no-stack-protector")
#pragma GCC target("sse,sse2,sse3,ssse3,sse4,popcnt,abm,mmx,avx,avx2")
#pragma GCC optimize("unroll-loops")*/
using ll = long long;
using ld = long double;
using namespace std;
#define ff first
#define ss second
#define sz(a) (int)(a).size()
#define forn(i, n) for (int i = 0; i < (int)n; ++i)
#define pb push_back
#define all(a) begin(a), end(a)
#ifdef LOCAL
#define debug(a) cerr << #a << ":\n" << a << endl
#else
#define debug(a)
#endif

mt19937 rnd(chrono::steady_clock::now().time_since_epoch().count());
const int INFi = 1e9 + 228;
const ll INFll = (ll)1e18 + 228;

template <typename T>
bool inmin(T& a, T b) {
    if (a <= b) return false;
    a = b; return true;
}

template <typename T>
bool inmax(T& a, T b) {
    if (a >= b) return false;
    a = b; return true;
}

void solve();

signed main() {
    ios_base::sync_with_stdio(0), cin.tie(0);
    cout << fixed << setprecision(10);
#ifdef LOCAL
    //freopen("/Users/alexsus/Desktop/solve/read.txt", "r", stdin);
#endif
    int tc;
    tc = 1;
    //cin >> tc;
    forn(i, tc) {
        solve();
    }
}

#define state pair<int, int>
vector<int> pr;
vector<int> a;
const int base = 51;
int dp[2][base * base * base][(1 << 3)];
pair<int, int> to[2][base * base * base][(1 << 3)];

int n;

vector<int> get_base(int num) {
    vector<int> res;
    forn(i, n) {
        res.pb(num % base);
        num /= base;
    }
    reverse(all(res));
    return res;
}

int calc(state now, int who) {
    if (dp[who][now.ff][now.ss] != -1) return dp[who][now.ff][now.ss];
```

```

    auto cur = get_base(now.ff);
    forn(i, n) {
        if (cur[i] > 0) {
            for (int j = 1; j <= cur[i]; ++j) {
                state adj = now;
                adj.ff -= j * pr[i];
                if (calc(adj, 1 - who) == who) {
                    dp[who][now.ff][now.ss] = who;
                    to[who][now.ff][now.ss] = {i, j};
                    return who;
                }
            }
        }
    }
    forn(i, n) {
        if (now.ss & (1 << i)) {
            state adj = now;
            adj.ff -= cur[i] * pr[i];
            adj.ff += a[i] * pr[i];
            adj.ss -= (1 << i);
            if (calc(adj, 1 - who) == who) {
                dp[who][now.ff][now.ss] = who;
                to[who][now.ff][now.ss] = {i, 0};
                return who;
            }
        }
    }
    dp[who][now.ff][now.ss] = 1 - who;
    return 1 - who;
}

void solve() {
    forn(i, 2) {
        forn(j, base * base * base) {
            forn(k, (1 << 3)) dp[i][j][k] = -1;
        }
    }
    cin >> n;
    pr.resize(n);
    pr[n - 1] = 1;
    for (int i = n - 2; i >= 0; --i) {
        pr[i] = pr[i + 1] * base;
    }
    a.resize(n);
    forn(i, n) cin >> a[i];
    state st;
    st.ff = 0;
    forn(i, n) {
        st.ff += a[i] * pr[i];
    }
    st.ss = (1 << n) - 1;
    int win = calc(st, 0);
    if (win == 1) {
        cout << -1 << '␣' << -1 << endl;
        return;
    }
    while (true) {
        auto move = to[0][st.ff][st.ss];
        auto cur = get_base(st.ff);
        cout << move.ff + 1 << '␣' << move.ss << endl;
        if (move.ss == 0) {
            st.ff -= cur[move.ff] * pr[move.ff];
            st.ff += a[move.ff] * pr[move.ff];
            st.ss -= (1 << move.ff);
        } else {
            st.ff -= move.ss * pr[move.ff];
        }
        cur = get_base(st.ff);
        pair<int, int> next;
        cin >> next.ff >> next.ss;
        if (next.ff == -1 && next.ss == -1) {
            return;
        }
    }
}

```

```

next.ff -= 1;
move = next;
if (move.ss == 0) {
    st.ff -= cur[move.ff] * pr[move.ff];
    st.ff += a[move.ff] * pr[move.ff];
    st.ss -= (1 << move.ff);
} else {
    st.ff -= move.ss * pr[move.ff];
}
}
}

```

Task E ()

```
#include "bits/stdc++.h"
/**pragma GCC optimize("Ofast,no-stack-protector")
#pragma GCC target("sse,sse2,sse3,ssse3,sse4,popcnt,abm,mmx,avx,avx2")
#pragma GCC optimize("unroll-loops")*/
using ll = long long;
using ld = long double;
using namespace std;
#define ff first
#define ss second
#define sz(a) (int)(a).size()
#define forn(i, n) for (int i = 0; i < (int)n; ++i)
#define pb push_back
#define all(a) begin(a), end(a)
#ifdef LOCAL
#define debug(a) cerr << #a << ":\n" << a << endl
#else
#define debug(a)
#endif

mt19937 rnd(chrono::steady_clock::now().time_since_epoch().count());
const int INFi = 1e9 + 228;
const ll INFll = (ll)1e18 + 228;

template <typename T>
bool inmin(T& a, T b) {
    if (a <= b) return false;
    a = b; return true;
}

template <typename T>
bool inmax(T& a, T b) {
    if (a >= b) return false;
    a = b; return true;
}

void solve();

signed main() {
    ios_base::sync_with_stdio(0), cin.tie(0);
    cout << fixed << setprecision(10);
#ifdef LOCAL
    freopen("/Users/alexsus/Desktop/solve/read.txt", "r", stdin);
#endif
    int tc;
    tc = 1;
    //cin >> tc;
    forn(i, tc) {
        solve();
    }
}

void solve() {
    int t;
    cin >> t;
    string wh;
    cin >> wh;
    vector<int> base;
    forn(_, t) {
        if (wh == "transmit") {
            int n;
            cin >> n;
            forn(i, 10) {
                int cnt = 0;
                if (n & (1 << i)) {
                    cnt = i + 1;
                }
                forn(j, 10) {
                    if (cnt) {
                        cout << '1';
                        cnt--;
                    } else {
                        cout << '0';
                    }
                }
            }
        }
    }
}
```

```

        }
    }
    cout << endl;
}
} else {
    int ans = 0;
    forn(i, 10) {
        int now = 0;
        forn(j, 10) {
            char x;
            cin >> x;
            now += x == '1';
        }
        if (now > 0) {
            ans += (1 << (now - 1));
        }
    }
    cout << ans << '\n';
}
}
}

```

Task F ()

```
#include "bits/stdc++.h"
/**pragma GCC optimize("Ofast,no-stack-protector")
#pragma GCC target("sse,sse2,sse3,ssse3,sse4,popcnt,abm,mmx,avx,avx2")
#pragma GCC optimize("unroll-loops")*/
using ll = long long;
using ld = long double;
using namespace std;
#define ff first
#define ss second
#define sz(a) (int)(a).size()
#define forn(i, n) for (int i = 0; i < (int)n; ++i)
#define pb push_back
#define all(a) begin(a), end(a)
#ifdef LOCAL
#define debug(a) cerr << #a << ":\n" << a << endl
#else
#define debug(a)
#endif

mt19937 rnd(chrono::steady_clock::now().time_since_epoch().count());
const int INFi = 1e9 + 228;
const ll INFll = (ll)1e18 + 228;

template <typename T>
bool inmin(T& a, T b) {
    if (a <= b) return false;
    a = b; return true;
}

template <typename T>
bool inmax(T& a, T b) {
    if (a >= b) return false;
    a = b; return true;
}

void solve();

signed main() {
    ios_base::sync_with_stdio(0), cin.tie(0);
    cout << fixed << setprecision(10);
#ifdef LOCAL
    freopen("/Users/alexsus/Desktop/solve/read.txt", "r", stdin);
#endif
    int tc;
    tc = 1;
    //cin >> tc;
    forn(i, tc) {
        solve();
    }
}

void to(string &s, vector<pair<int, ll>> &res) {
    int n = sz(s);
    for (int i = 0; i < n; i++) {
        if (s[i] != '(') {
            res.pb({s[i] - '0', 1});
            ++i;
        } else {
            ++i;
            int num = s[i] - '0';
            ll cnt = 0;
            i += 2;
            while (s[i] != ')') {
                cnt *= 10;
                cnt += s[i] - '0';
                ++i;
            }
            ++i;
            if (cnt > 0)
                res.pb({num, cnt});
        }
    }
}
```

```

    reverse(all(res));
}

const ll INFF = 1e10;

void solve() {
    string s, t;
    cin >> s >> t;
    vector<pair<int, ll>> sa, sb;
    to(s, sa);
    to(t, sb);
    if (sb.empty()) {
        cout << s << '\n';
        return;
    } else if (sa.empty()) {
        cout << t << '\n';
        return;
    }
    vector<pair<int, ll>> ans;
    int it1 = 0, it2 = 0;
    auto last1 = sa[it1];
    auto last2 = sb[it2];
    int rem = 0;
    pair<int, ll> zr = {0, INFF};
    while (true) {
        if (last1.ff == 0 && last1.ss > INFF && last2.ff == 0 && last2.ss > INFF && rem == 0)
            break;
        auto [x, c1] = last1;
        auto [y, c2] = last2;
        ll mn = min(c1, c2);
        if (c1 < c2) {
            last2 = {y, c2 - c1};
            if (it1 + 1 < sz(sa)) {
                last1 = sa[++it1];
            } else {
                last1 = zr;
            }
        } else if (c1 > c2) {
            last1 = {x, c1 - c2};
            if (it2 + 1 < sz(sb)) {
                last2 = sb[++it2];
            } else {
                last2 = zr;
            }
        } else {
            if (it1 + 1 < sz(sa)) {
                last1 = sa[++it1];
            } else {
                last1 = zr;
            }
            if (it2 + 1 < sz(sb)) {
                last2 = sb[++it2];
            } else {
                last2 = zr;
            }
        }
    }
    if (rem == 0) {
        if (x + y < 10) {
            ans.pb({x + y, mn});
        } else {
            int val = (x + y) % 10;
            ans.pb({val, 1});
            rem = 1;
            if (mn - 1 > 0) {
                ans.pb({val + 1, mn - 1});
            }
        }
    } else {
        if (x + y + rem < 10) {
            ans.pb({x + y + rem, 1});
            rem = 0;
            if (mn - 1 > 0) {
                ans.pb({x + y, mn - 1});
            }
        }
    }
}

```

```

        } else {
            int val = (x + y + rem) % 10;
            ans.pb({val, mm});
            rem = 1;
        }
    }
}
while (ans.back().ff == 0) ans.pop_back();
reverse(all(ans));
for (auto [x, c] : ans) {
    cout << "(" << x << "|" << c << ")";
}
}

```