

Олимпиада СПбГУ по информатике 2022/23 учебного года

A	B	C	D	E	F	Sum
100	100	40	100	16	10	366

Task A ()

```
#include <bits/stdc++.h>
// tiom4eg's precompiler options
// POGGERS POGGERS POGGERS POGGERS POGGERS POGGERS POGGERS
// IO settings
#define fastIO ios_base::sync_with_stdio(false); cin.tie(0)
// Quick types
#define ll long long
#define ld long double
#define ull unsigned long long
#define pii pair<int, int>
#define vi vector<int>
#define mi vector<vector<int>>
// Quick functions
#define endl "\n"
#define F first
#define S second
#define all(a) a.begin(), a.end()
#define sz(a) (int)(a.size())
#define pb push_back
#define mp make_pair
// Quick fors
#define FOR(i, a, b) for (int i = a; i < b; ++i)
#define FORS(i, a, b, c) for (int i = a; i < b; i += c)
#define RFOR(i, a, b) for (int i = a; i >= b; --i)
#define EACH(e, a) for (auto& e : a)
// Pragmas
#ifndef TIOM4EG
#pragma GCC optimize("O3,unroll-loops") // let the chaos begin!
#pragma GCC target("avx,avx2,bmi,bmi2,tune=native")
#pragma GCC comment(linker, "/stack:200000000")
#endif
// PBDs
#include <ext/pb_ds/assoc_container.hpp>
#include <ext/pb_ds/tree_policy.hpp>
#define pbds tree<int, null_type, less_equal<int>, rb_tree_tag,
    tree_order_statistics_node_update>
#define ook order_of_key
#define fbo find_by_order
using namespace __gnu_pbds;
// POGGERS POGGERS POGGERS POGGERS POGGERS POGGERS POGGERS
using namespace std;
mt19937 rng(chrono::duration_cast<chrono::milliseconds>(chrono::system_clock::now()).
    time_since_epoch()).count();
// #define int long long
const int INF = 1e9 + 7, MD = 998244353, MAX = 100005, R = 1 << 18, MOD = 1e9 + 7, MOD2 = 1e9 + 9,
    LG = 18;
const ll INFLL = 1e18 + 7;

signed main() {
    fastIO;
    int a[6];
    FOR(i, 0, 6) {
        cin >> a[i];
        FOR(j, 0, i) if (a[j] >= a[i]) ++a[j];
    }
    FOR(i, 0, 6) cout << a[i] << ' ';
}
```

}

Task B ()

```

#include <bits/stdc++.h>
// tiom4eg's precompiler options
// POGGERS POGGERS POGGERS POGGERS POGGERS POGGERS POGGERS
// IO settings
#define fastIO ios_base::sync_with_stdio(false); cin.tie(0)
// Quick types
#define ll long long
#define ld long double
#define ull unsigned long long
#define pii pair<int, int>
#define vi vector<int>
#define mi vector<vector<int>>
// Quick functions
#define endl "\n"
#define F first
#define S second
#define all(a) a.begin(), a.end()
#define sz(a) (int)(a.size())
#define pb push_back
#define mp make_pair
// Quick fors
#define FOR(i, a, b) for (int i = a; i < b; ++i)
#define FORS(i, a, b, c) for (int i = a; i < b; i += c)
#define RFOR(i, a, b) for (int i = a; i >= b; --i)
#define EACH(e, a) for (auto& e : a)
// Pragmas
#ifndef TIOM4EG
#pragma GCC optimize("O3,unroll-loops") // let the chaos begin!
#pragma GCC target("avx,avx2,bmi,bmi2,tune=native")
#pragma GCC comment(linker, "/stack:200000000")
#endif
// PBDS
#include <ext/pb_ds/assoc_container.hpp>
#include <ext/pb_ds/tree_policy.hpp>
#define pbds tree<int, null_type, less_equal<int>, rb_tree_tag,
    tree_order_statistics_node_update>
#define ook order_of_key
#define fbo find_by_order
using namespace __gnu_pbds;
// POGGERS POGGERS POGGERS POGGERS POGGERS POGGERS POGGERS
using namespace std;
mt19937 rng(chrono::duration_cast<chrono::milliseconds>(chrono::system_clock::now()).
    time_since_epoch()).count());
#define int long long
const int INF = 1e9 + 7, MD = 998244353, MAX = 100005, R = 1 << 18, MOD = 1e9 + 7, MOD2 = 1e9 + 9,
    LG = 18;
const ll INFLL = 1e18 + 7;

signed main() {
    fastIO;
    string run; cin >> run;
    if (run[0] == 'f') {
        int n, s = 0, x; cin >> n;
        FOR(i, 0, n) cin >> x, s += x;
        cout << 1000 * s;
    } else {
        int n, s = 0, p, x; cin >> n;
        cin >> x;
        p = x / 1000, s += x % 1000;
        FOR(i, 1, n) cin >> x, s += x % 1000;
        cout << s + p;
    }
}

```

Task C ()

```

#include <bits/stdc++.h>
// tiom4eg's precompiler options
// POGGERS POGGERS POGGERS POGGERS POGGERS POGGERS POGGERS
// IO settings
#define fastIO ios_base::sync_with_stdio(false); cin.tie(0)
// Quick types
#define ll long long
#define ld long double
#define ull unsigned long long
#define pii pair<int, int>
#define vi vector<int>
#define mi vector<vector<int>>
// Quick functions
// #define endl "\n"
#define F first
#define S second
#define all(a) a.begin(), a.end()
#define sz(a) (int)(a.size())
#define pb push_back
#define mp make_pair
// Quick fors
#define FOR(i, a, b) for (int i = a; i < b; ++i)
#define FORS(i, a, b, c) for (int i = a; i < b; i += c)
#define RFOR(i, a, b) for (int i = a; i >= b; --i)
#define EACH(e, a) for (auto& e : a)
// Pragmas
#ifdef TIOM4EG
#pragma GCC optimize("O3,unroll-loops") // let the chaos begin!
#pragma GCC target("avx,avx2,bmi,bmi2,tune=native")
#pragma GCC comment(linker, "/stack:200000000")
#endif
// PBDS
#include <ext/pb_ds/assoc_container.hpp>
#include <ext/pb_ds/tree_policy.hpp>
#define pbds tree<int, null_type, less_equal<int>, rb_tree_tag,
    tree_order_statistics_node_update>
#define ook order_of_key
#define fbo find_by_order
using namespace __gnu_pbds;
// POGGERS POGGERS POGGERS POGGERS POGGERS POGGERS POGGERS
using namespace std;
mt19937 rng(chrono::duration_cast<chrono::milliseconds>(chrono::system_clock::now().
    time_since_epoch()).count());
#define int long long
const int INF = 1e9 + 7, MD = 998244353, MAX = 100005, R = 1 << 18, MOD = 1e9 + 7, MOD2 = 1e9 + 9,
    LG = 18;
const ll INFLL = 1e18 + 7;

void chmin(int& a, int b) { if (a > b) a = b; }
void chmax(int& a, int b) { if (a < b) a = b; }

const int dx[] = {-1, 0, 1, 0}, dy[] = {0, -1, 0, 1};

signed main() {
    fastIO;
    set<pii> res;
    pii z[3];
    FOR(i, 0, 3) cin >> z[i].F >> z[i].S, res.insert({min(z[i].F, z[i].S), max(z[i].F, z[i].S)});
    vi o = {0, 1, 2};
    do {
        FOR(ori1, 0, 2) {
            swap(z[o[0]].F, z[o[0]].S);
            FOR(ori2, 0, 2) {
                swap(z[o[1]].F, z[o[1]].S);
                FOR(x1, 0, z[o[0]].F - z[o[1]].F + 1) FOR(y1, 0, z[o[0]].S - z[o[1]].S + 1) if ((
                    x1 == 0 || x1 == z[o[0]].F - z[o[1]].F) && (y1 == 0 || y1 == z[o[0]].S - z[o
                    [1]].S)) {
                    vector<vi> comp(z[o[0]].F, vi(z[o[0]].S));
                    FOR(gg, x1, x1 + z[o[1]].F) FOR(hh, y1, y1 + z[o[1]].S) comp[gg][hh] = 1;
                    ///
                    vector<vi> used(z[o[0]].F, vi(z[o[0]].S));

```


Task D ()

```

#include <bits/stdc++.h>
// tiom4eg's precompiler options
// POGGERS POGGERS POGGERS POGGERS POGGERS POGGERS POGGERS
// IO settings
#define fastIO ios_base::sync_with_stdio(false); cin.tie(0)
// Quick types
#define ll long long
#define ld long double
#define ull unsigned long long
#define pii pair<int, int>
#define vi vector<int>
#define mi vector<vector<int>>
// Quick functions
// #define endl "\n"
#define F first
#define S second
#define all(a) a.begin(), a.end()
#define sz(a) (int)(a.size())
#define pb push_back
#define mp make_pair
// Quick fors
#define FOR(i, a, b) for (int i = a; i < b; ++i)
#define FORS(i, a, b, c) for (int i = a; i < b; i += c)
#define RFOR(i, a, b) for (int i = a; i >= b; --i)
#define EACH(e, a) for (auto& e : a)
// Pragmas
#ifndef TIOM4EG
#pragma GCC optimize("O3,unroll-loops") // let the chaos begin!
// #pragma GCC target("avx,tune=native")
// #pragma GCC comment(linker, "/stack:200000000")
#endif
// PBDs
/* #include <ext/pb_ds/assoc_container.hpp>
#include <ext/pb_ds/tree_policy.hpp>
#define pbds tree<int, null_type, less_equal<int>, rb_tree_tag,
tree_order_statistics_node_update>
#define ook order_of_key
#define fbo find_by_order
using namespace __gnu_pbds; */
// POGGERS POGGERS POGGERS POGGERS POGGERS POGGERS POGGERS
using namespace std;
mt19937 rng(chrono::duration_cast<chrono::milliseconds>(chrono::system_clock::now()).
time_since_epoch()).count());
// #define int long long
const int INF = 1e9 + 7, MD = 998244353, MAX = 100005, R = 1 << 18, MOD = 1e9 + 7, MOD2 = 1e9 + 9,
LG = 18;
const ll INFLL = 1e18 + 7;
const int FAC[] = {41616, 816, 16, 8, 4, 2, 1};

int n, hh;
unordered_map<int, pii> dp;
array<int, 7> a, cur;
pii res;

pii calc() {
    if (dp.find(hh) != dp.end()) return dp[hh];
    if (cur[6]) {
        FOR(i, 0, n) {
            if (!cur[3 + i]) {
                int x = cur[i];
                hh = hh + (a[i] - x) * FAC[i] + FAC[3 + i] - FAC[6];
                cur[i] = a[i], cur[3 + i] = 1, cur[6] = 0;
                res = calc();
                cur[i] = x, cur[3 + i] = 0, cur[6] = 1;
                hh = hh - (a[i] - x) * FAC[i] - FAC[3 + i] + FAC[6];
                if (res.F < 0) return dp[hh] = {-1, -1};
            }
        }
        FOR(j, 1, cur[i] + 1) {
            hh = hh - j * FAC[i] - FAC[6];
            cur[i] -= j, cur[6] = 0;
            res = calc();
            cur[i] += j, cur[6] = 1;
        }
    }
}

```

```

        hh = hh + j * FAC[i] + FAC[6];
        if (res.F < 0) return dp[hh] = {-1, -1};
    }
}
return dp[hh] = {0, -1};
} else {
    FOR(i, 0, n) {
        if (!cur[3 + i]) {
            int x = cur[i];
            hh = hh + (a[i] - x) * FAC[i] + FAC[3 + i] + FAC[6];
            cur[i] = a[i], cur[3 + i] = 1, cur[6] = 1;
            res = calc();
            cur[i] = x, cur[3 + i] = 0, cur[6] = 0;
            hh = hh - (a[i] - x) * FAC[i] - FAC[3 + i] - FAC[6];
            if (res.F >= 0) return dp[hh] = {i, 0};
        }
        FOR(j, 1, cur[i] + 1) {
            hh = hh - j * FAC[i] + FAC[6];
            cur[i] -= j, cur[6] = 1;
            res = calc();
            cur[i] += j, cur[6] = 0;
            hh = hh + j * FAC[i] - FAC[6];
            if (res.F >= 0) return dp[hh] = {i, j};
        }
    }
    return dp[hh] = {-1, -1};
}
}

int hashme(array<int, 7> a) {
    int h = 0;
    FOR(i, 0, 7) h += FAC[i] * a[i];
    return h;
}

signed main() {
    fastIO;
    cin >> n;
    FOR(i, 0, n) cur[3 + i] = 1;
    dp[hashme(cur)] = {-1, -1}; // lose
    cur[6] = 1;
    dp[hashme(cur)] = {0, -1}; // win
    cur[6] = 0;
    FOR(i, 0, n) cur[3 + i] = 0;
    FOR(i, 0, n) cin >> cur[i], a[i] = cur[i];
    hh = hashme(cur);
    calc();
    if (dp[hh].F == -1) return !(cout << "-1_-1" << endl);
    while (true) {
        hh = hashme(cur);
        if (dp.find(hh) == dp.end()) calc();
        pii now = dp[hh];
        cout << now.F + 1 << '┐' << now.S << endl;
        if (!now.S) cur[now.F] = a[now.F], cur[3 + now.F] = 1;
        else cur[now.F] -= now.S;
        int p, c; cin >> p >> c, --p;
        if (c == -1) return 0;
        if (!c) cur[p] = a[p], cur[3 + p] = 1;
        else cur[p] -= c;
    }
}

```

Task E ()

```

#include <bits/stdc++.h>
// tiom4eg's precompiler options
// POGGERS POGGERS POGGERS POGGERS POGGERS POGGERS POGGERS
// IO settings
#define fastIO ios_base::sync_with_stdio(false); cin.tie(0)
// Quick types
#define ll long long
#define ld long double
#define ull unsigned long long
#define pii pair<int, int>
#define vi vector<int>
#define mi vector<vector<int>>
// Quick functions
#define endl "\n"
#define F first
#define S second
#define all(a) a.begin(), a.end()
#define sz(a) (int)(a.size())
#define pb push_back
#define mp make_pair
// Quick fors
#define FOR(i, a, b) for (int i = a; i < b; ++i)
#define FORS(i, a, b, c) for (int i = a; i < b; i += c)
#define RFOR(i, a, b) for (int i = a; i >= b; --i)
#define EACH(e, a) for (auto& e : a)
// Pragmas
#ifdef TIOM4EG
#pragma GCC optimize("O3,unroll-loops") // let the chaos begin!
#pragma GCC target("avx,avx2,bmi,bmi2,tune=native")
#pragma GCC comment(linker, "/stack:200000000")
#endif
// PBDs
#include <ext/pb_ds/assoc_container.hpp>
#include <ext/pb_ds/tree_policy.hpp>
#define pbds tree<int, null_type, less_equal<int>, rb_tree_tag,
    tree_order_statistics_node_update>
#define ook order_of_key
#define fbo find_by_order
using namespace __gnu_pbds;
// POGGERS POGGERS POGGERS POGGERS POGGERS POGGERS POGGERS
using namespace std;
mt19937 rng(chrono::duration_cast<chrono::milliseconds>(chrono::system_clock::now()).
    time_since_epoch()).count();
// #define int long long
const int INF = 1e9 + 7, MD = 998244353, MAX = 100005, R = 1 << 18, MOD = 1e9 + 7, MOD2 = 1e9 + 9,
    LG = 18;
const ll INFLL = 1e18 + 7;

array<int, 10> x;
vector<array<int, 10>> ok;

void precalc(int p, int l) {
    if (p == 10) {
        ok.pb(x);
        return;
    }
    FOR(i, l, 11) {
        x[p] = i;
        precalc(p + 1, i);
    }
}

void TLE() {
    int x = 0;
    FOR(i, 0, 1000000000LL) x = 2 * x * x * x + 3 * x * x + 4 * x + 5;
}

signed main() {
    precalc(0, 0);
    fastIO;
    int tc; cin >> tc;
    string run; cin >> run;
}

```



```

if (run[0] == 't') {
    while (tc--) {
        int x; cin >> x, --x;
        if (x > 184755) TLE();
        FOR(i, 0, 10) {
            FOR(j, 0, ok[x][i]) cout << '1';
            FOR(j, 0, 10 - ok[x][i]) cout << '0';
            cout << endl;
        }
        cout << endl;
    }
} else {
    while (tc--) {
        array<int, 10> c{};
        FOR(i, 0, 10) {
            string s; cin >> s;
            FOR(j, 0, 10) c[i] += (s[j] == '1');
        }
        sort(all(c));
        cout << (lower_bound(all(ok), c) - ok.begin()) + 1 << endl;
    }
}
}

```

Task F ()

```

#include <bits/stdc++.h>
// tiom4eg's precompiler options
// POGGERS POGGERS POGGERS POGGERS POGGERS POGGERS POGGERS
// IO settings
#define fastIO ios_base::sync_with_stdio(false); cin.tie(0)
// Quick types
#define ll long long
#define ld long double
#define ull unsigned long long
#define pii pair<int, int>
#define vi vector<int>
#define mi vector<vector<int>>
// Quick functions
// #define endl "\n"
#define F first
#define S second
#define all(a) a.begin(), a.end()
#define sz(a) (int)(a.size())
#define pb push_back
#define mp make_pair
// Quick fors
#define FOR(i, a, b) for (int i = a; i < b; ++i)
#define FORS(i, a, b, c) for (int i = a; i < b; i += c)
#define RFOR(i, a, b) for (int i = a; i >= b; --i)
#define EACH(e, a) for (auto& e : a)
// Pragmas
#ifdef TIOM4EG
#pragma GCC optimize("O3,unroll-loops") // let the chaos begin!
#pragma GCC target("avx,avx2,bmi,bmi2,tune=native")
#pragma GCC comment(linker, "/stack:200000000")
#endif
// PBDs
#include <ext/pb_ds/assoc_container.hpp>
#include <ext/pb_ds/tree_policy.hpp>
#define pbds tree<int, null_type, less_equal<int>, rb_tree_tag,
    tree_order_statistics_node_update>
#define ook order_of_key
#define fbo find_by_order
using namespace __gnu_pbds;
// POGGERS POGGERS POGGERS POGGERS POGGERS POGGERS POGGERS
using namespace std;
mt19937 rng(chrono::duration_cast<chrono::milliseconds>(chrono::system_clock::now()).
    time_since_epoch()).count();
#define int long long
const int INF = 1e9 + 7, MD = 998244353, MAX = 100005, R = 1 << 18, MOD = 1e9 + 7, MOD2 = 1e9 + 9,
    LG = 18;
const ll INFLL = 1e18 + 7;

signed main() {
    fastIO;
    string a, b; cin >> a >> b;
    vector<pii> ba, bb;
    for (int p = 0; p < sz(a);) {
        if (a[p] == '(') {
            int num = 0, cnt = 0;
            while (a[++p] != '|') num = 10 * num + (a[p] - '0');
            while (a[++p] != ')') cnt = 10 * cnt + (a[p] - '0');
            ba.pb({num, cnt});
        }
        else ba.pb({a[p] - '0', 1});
        ++p;
    }
    for (int p = 0; p < sz(b);) {
        if (b[p] == '(') {
            int num = 0, cnt = 0;
            while (b[++p] != '|') num = 10 * num + (b[p] - '0');
            while (b[++p] != ')') cnt = 10 * cnt + (b[p] - '0');
            bb.pb({num, cnt});
        }
        else bb.pb({b[p] - '0', 1});
        ++p;
    }
}

```

```

reverse(all(ba)), reverse(all(bb));
vector<pii> ev;
int lena = 0, lenb = 0;
FOR(i, 0, sz(ba)) {
    lena += ba[i].S;
    ev.pb({lena, 0});
}
FOR(i, 0, sz(bb)) {
    lenb += bb[i].S;
    ev.pb({lenb, 1});
}
ba.pb({0, 0}), bb.pb({0, 0});
/*cout << "----\n";
EACH(e, ba) cout << e.F << ' ' << e.S << endl;
cout << "----\n";
EACH(e, bb) cout << e.F << ' ' << e.S << endl;
cout << "----\n";*/
ev.pb({0, -1});
ev.pb({max(lena, lenb) + 1, -1});
sort(all(ev));
/*EACH(e, ev) cout << e.F << ' ' << e.S << endl;
cout << "----\n";*/
vector<pii> res;
int add = 0, ptr = 0, ptrb = 0;
FOR(i, 1, sz(ev)) {
    if (ev[i].F - ev[i - 1].F) {
        if (add) {
            if (ba[ptr].F + bb[ptrb].F > 8) {
                res.pb({ba[ptr].F + bb[ptrb].F - 9, ev[i].F - ev[i - 1].F});
            } else {
                res.pb({ba[ptr].F + bb[ptrb].F + 1, 1});
                res.pb({ba[ptr].F + bb[ptrb].F, ev[i].F - ev[i - 1].F - 1});
                add = 0;
            }
        } else {
            if (ba[ptr].F + bb[ptrb].F > 9) {
                res.pb({ba[ptr].F + bb[ptrb].F - 10, 1});
                res.pb({ba[ptr].F + bb[ptrb].F - 9, ev[i].F - ev[i - 1].F - 1});
                add = 1;
            } else {
                res.pb({ba[ptr].F + bb[ptrb].F, ev[i].F - ev[i - 1].F});
            }
        }
    }
    if (ev[i].S == 0) ptr++;
    else if (ev[i].S == 1) ptrb++;
}
reverse(all(res));
EACH(e, res) if (e.S) cout << '(' << e.F << '|' << e.S << ')';
}

```