

Олимпиада СПбГУ по информатике 2022/23 учебного года

A	B	C	D	E	F	Sum
100	100	90	60	52	65	467

Task A ()

```
// Created by a on 06.03.2022.
//
// clang-format off

// #include <random>

#include <algorithm>
#include <unordered_set>
#pragma GCC target ("sse, sse2, sse3, ssse3, popcnt, abm, mmx, tune=native")
#pragma GCC optimize ("fast-math")
#pragma GCC optimize ("O3", "no-stack-protector", "inline", "unroll-loops")
#pragma comment(linker, "/stack:200000000")
// #pragma GCC target ("avx, avx2, fma")

#include <bits/stdc++.h>
#include <ext/rope>

// #include <ext/pb_ds/assoc_container.hpp>
// #include <ext/pb_ds/tree_policy.hpp>
// using namespace __gnu_pbds;

// #define ordered_set tree<int, null_type, less<int>, rb_tree_tag,
//     tree_order_statistics_node_update>

#define endl "\n"
// #define int int_fast32_t
#define int long long
#define ui unsigned int
#define all(a) a.begin(), a.end()
#define rall(a) a.rbegin(), a.rend()
#define mall(a) a.mutable_begin(), a.mutable_end()
#define uni(x) sort(all(x)); x.erase(unique(all(x)), x.end())
#define X first
#define Y second
// #define short unsigned short
#define pii pair<int, int>
#define pb push_back
#define base complex<double>
#define ll long long
#define ull unsigned ll
// #define ui int_fast64_t
// #define double long double
#define fast ios_base::sync_with_stdio(0); cin.tie(0); cout.tie(0)

using namespace std;
using namespace __gnu_cxx;

template<typename T, typename G>
bool rmax(T &a, G b1) {
    T b = static_cast<T>(b1);
    if (a < b) {
        a = b;
        return 1;
    }
    return 0;
}
```

```

}

template<typename T, typename G>
bool rmin(T &a, G b) {
    T b1 = static_cast<T>(b);
    if (b1 < a) {
        a = b1;
        return 1;
    }
    return 0;
}

template<typename T>
istream &operator>>(istream &in, vector<T> &a) {
    for (auto &it : a) {
        in >> it;
    }
    return in;
}

template<typename T>
ostream &operator<<(ostream &out, vector<T> &a) {
    for (auto &it : a) {
        out << it << ' ';
    }
    return out;
}

void solve() {
    int n = 6;

    rope<int> my;

    for (int i = 0; i < n; i++) {
        int num;
        cin >> num;
        num--;

        auto l = my.substr(0, num);
        l.push_back(i);
        my.erase(0, num);
        my = l + my;
        // }
    }

    vector<int> id(n);
    for (int i = 0; i < my.size(); i++) {
        id[my[i]] = i + 1;
    }

    cout << id << endl;
}

signed main() {
    fast;
    // cout.precision(10);
    // cout << fixed;

    #if defined(LOCAL_CLION)
        freopen("../input.txt", "r", stdin);
        freopen("../output.txt", "w", stdout);
    #elif defined(LOCAL)
        freopen("../input.txt", "r", stdin);
        freopen("../output.txt", "w", stdout);
    #else
        // freopen("../mining.in", "r", stdin);
        // freopen("../mining.out", "w", stdout);
    #endif

    int t = 1;
    // cin >> t;

```

```
    while (t--)  
        solve();  
}  
  
//clang-format on
```

Task B ()

```
// Created by a on 06.03.2022.
//
// clang-format off

// #include <random>

#include <algorithm>
#include <numeric>
#include <unordered_set>
#pragma GCC target("sse,sse2,sse3,ssse3,popcnt,abm,mmx,tune=native")
#pragma GCC optimize("fast-math")
#pragma GCC optimize("O3", "no-stack-protector", "inline", "unroll-loops")
#pragma comment(linker, "/stack:200000000")
// #pragma GCC target("avx,avx2,fma")

#include <bits/stdc++.h>
#include <ext/rope>

// #include <ext/pb_ds/assoc_container.hpp>
// #include <ext/pb_ds/tree_policy.hpp>
// using namespace __gnu_pbds;

// #define ordered_set tree<int, null_type, less<int>, rb_tree_tag,
//     tree_order_statistics_node_update>

#define endl "\n"
// #define int int_fast32_t
#define int long long
#define ui unsigned int
#define all(a) a.begin(), a.end()
#define rall(a) a.rbegin(), a.rend()
#define mall(a) a.mutable_begin(), a.mutable_end()
#define uni(x) sort(all(x)); x.erase(unique(all(x)), x.end())
#define X first
#define Y second
// #define short unsigned short
#define pii pair<int, int>
#define pb push_back
#define base complex<double>
#define ll long long
#define ull unsigned ll
// #define ui int_fast64_t
// #define double long double
#define fast ios_base::sync_with_stdio(0); cin.tie(0); cout.tie(0)

using namespace std;
using namespace __gnu_cxx;

template<typename T, typename G>
bool rmax(T &a, G b1) {
    T b = static_cast<T>(b1);
    if (a < b) {
        a = b;
        return 1;
    }
    return 0;
}

template<typename T, typename G>
bool rmin(T &a, G b) {
    T b1 = static_cast<T>(b);
    if (b1 < a) {
        a = b1;
        return 1;
    }
    return 0;
}

template<typename T>
istream &operator>>(istream &in, vector<T> &a) {
    for (auto &it : a) {
```

```

        in >> it;
    }
    return in;
}

template<typename T>
ostream &operator<<(ostream &out, vector<T> &a) {
    for (auto &it : a) {
        out << it << ' ';
    }
    return out;
}

void solve() {
    string f;
    cin >> f;

    if (f == "first") {
        int n;
        cin >> n;

        vector<int> a(n);
        cin >> a;

        int sm = 0;
        for (int &it : a) {
            sm += it;
        }

        cout << sm * 101 << endl;
        return;
    }

    int n;
    cin >> n;
    vector<int> a(n);
    cin >> a;

    int b1 = a[0] % 101;

    int x = a[0] / 101;
    int sum = x;
    for (int &it : a) {
        sum += it % 101;;
    }

    cout << sum << endl;
    return;
    int sm = 0;
    for (int &it : a) {
        sm += it;
    }

    cout << sm << endl;
}

signed main() {
    fast;
    // cout.precision(10);
    // cout << fixed;

    #if defined(LOCAL_CLION)
        freopen("../input.txt", "r", stdin);
        freopen("../output.txt", "w", stdout);
    #elif defined(LOCAL)
        freopen("../input.txt", "r", stdin);
        freopen("../output.txt", "w", stdout);
    #else
        // freopen("../mining.in", "r", stdin);
        // freopen("../mining.out", "w", stdout);
    #endif
}

```

```
int t = 1;
// cin >> t;

while (t--)
    solve();
}

//clang-format on
```

Task C ()

```
// Created by a on 06.03.2022.
//
// clang-format off

// #include <random>

#include <algorithm>
#include <numeric>
#include <unordered_set>
#pragma GCC target("sse,sse2,sse3,ssse3,popcnt,abm,mmx,tune=native")
#pragma GCC optimize("fast-math")
#pragma GCC optimize("O3", "no-stack-protector", "inline", "unroll-loops")
#pragma comment(linker, "/stack:200000000")
// #pragma GCC target("avx,avx2,fma")

#include <bits/stdc++.h>
#include <ext/rope>

// #include <ext/pb_ds/assoc_container.hpp>
// #include <ext/pb_ds/tree_policy.hpp>
// using namespace __gnu_pbds;

// #define ordered_set tree<int, null_type, less<int>, rb_tree_tag,
//     tree_order_statistics_node_update>

#define endl "\n"
// #define int int_fast32_t
#define int long long
#define ui unsigned int
#define all(a) a.begin(), a.end()
#define rall(a) a.rbegin(), a.rend()
#define mall(a) a.mutable_begin(), a.mutable_end()
#define uni(x) sort(all(x)); x.erase(unique(all(x)), x.end())
#define X first
#define Y second
// #define short unsigned short
#define pii pair<int, int>
#define pb push_back
#define base complex<double>
#define ll long long
#define ull unsigned ll
// #define ui int_fast64_t
// #define double long double
#define fast ios_base::sync_with_stdio(0); cin.tie(0); cout.tie(0)

using namespace std;
using namespace __gnu_cxx;

template<typename T, typename G>
bool rmax(T &a, G b1) {
    T b = static_cast<T>(b1);
    if (a < b) {
        a = b;
        return 1;
    }
    return 0;
}

template<typename T, typename G>
bool rmin(T &a, G b) {
    T b1 = static_cast<T>(b);
    if (b1 < a) {
        a = b1;
        return 1;
    }
    return 0;
}

template<typename T>
istream &operator>>(istream &in, vector<T> &a) {
    for (auto &it : a) {
```

```

        in >> it;
    }
    return in;
}

template<typename T>
ostream &operator<<(ostream &out, vector<T> &a) {
    for (auto &it : a) {
        out << it << ' ';
    }
    return out;
}

bool is_in(pii a, pii b) {
    return (a.X >= b.X && a.Y >= b.Y);
}

void solve() {
    pii a, b, c;

    cin >> a.X >> a.Y;
    cin >> b.X >> b.Y;
    cin >> c.X >> c.Y;

    vector<pii> ans;

    ans.push_back(a);
    ans.push_back(b);
    ans.push_back(c);

    auto calc = [&] (pii a, pii b, pii c) {
        if (!is_in(a, b)) {
            return;
        }

        if (a.X == b.X) {
            ans.emplace_back(a.X, a.Y - b.Y);
        }

        if (!is_in(a, c)) {
            return;
        }

        if (b.Y + c.Y == a.Y && a.X == b.X + c.X) {
            ans.emplace_back(a.X - b.X, b.Y);
        }

        if (a.X == b.X && a.X == c.X && a.Y > c.Y + b.Y) {
            ans.emplace_back(a.X, a.Y - c.Y - b.Y);
        }

        if (a.X == b.X + c.X && c.Y == a.Y) {
            ans.emplace_back(b.X, a.Y - b.Y);
        }

        if (a.X == c.X && b.Y + c.Y < a.Y) {
            ans.emplace_back(a.X, a.Y - c.Y - b.Y);
            ans.emplace_back(a.X - b.X, b.Y);
        }

        if (b.Y + c.Y == a.Y && a.X <= b.X + c.X) {
            ans.emplace_back(a.X - c.X, c.Y);
        }

        if (b.X == c.X && a.Y == b.Y + c.Y) {
            ans.emplace_back(a.X - b.X, a.Y);
        }

        if (b.X + c.X <= a.X && b.Y + c.Y >= a.Y) {
            ans.emplace_back(b.X, a.Y - b.Y);
        }
    };
}

```



```

vector<pii> ad = {a, b, c};
vector<int> perm = {0, 1, 2};

do {
    vector<pii> new_ad;
    for (int &it : perm) {
        new_ad.emplace_back(ad[it]);
    }
    int n = ad.size();
    for (int mask = 0; mask < (1 << n); mask++) {
        for (int i = 0; i < n; i++) {
            if (mask & (1 << i)) {
                swap(new_ad[i].X, new_ad[i].Y);
            }
        }

        calc(new_ad[0], new_ad[1], new_ad[2]);

        for (int i = 0; i < n; i++) {
            if (mask & (1 << i)) {
                swap(new_ad[i].X, new_ad[i].Y);
            }
        }
    }
} while (next_permutation(all(perm)));

for (pii &it : ans) {
    // it.X = abs(it.X);
    // it.Y = abs(it.Y);
    if (it.X > it.Y) {
        swap(it.X, it.Y);
    }
}

uni(ans);

for (pii &it : ans) {
    if (it.X < 0 || it.Y < 0) {
        continue;
    }
    if (it.X * it.Y == 0) {
        continue;
    }
    cout << it.X << ' ' << it.Y << endl;
}

}

signed main() {
    fast;
    // cout.precision(10);
    // cout << fixed;

    #if defined(LOCAL_CLION)
        freopen("../input.txt", "r", stdin);
        freopen("../output.txt", "w", stdout);
    #elif defined(LOCAL)
        freopen("./input.txt", "r", stdin);
        freopen("./output.txt", "w", stdout);
    #else
        // freopen("./mining.in", "r", stdin);
        // freopen("./mining.out", "w", stdout);
    #endif

    int t = 1;
    // cin >> t;

    while (t--)
        solve();
}

//clang-format on

```

Task D ()

```
// Created by a on 06.03.2022.
//
// clang-format off

// #include <random>

#include <algorithm>
#include <numeric>
#include <string>
#include <unordered_set>
#pragma GCC target ("sse, sse2, sse3, ssse3, popcnt, abm, mmx, tune=native")
#pragma GCC optimize ("fast-math")
#pragma GCC optimize ("O3", "no-stack-protector", "inline", "unroll-loops")
#pragma comment(linker, "/stack:200000000")
// #pragma GCC target ("avx, avx2, fma")

#include <bits/stdc++.h>
#include <ext/rope>

// #include <ext/pb_ds/assoc_container.hpp>
// #include <ext/pb_ds/tree_policy.hpp>
// using namespace __gnu_pbds;

// #define ordered_set tree<int, null_type, less<int>, rb_tree_tag,
//     tree_order_statistics_node_update>

// #define endl "\n"
// #define int int_fast32_t
#define int long long
#define ui unsigned int
#define all(a) a.begin(), a.end()
#define rall(a) a.rbegin(), a.rend()
#define mall(a) a.mutable_begin(), a.mutable_end()
#define uni(x) sort(all(x)); x.erase(unique(all(x)), x.end())
#define X first
#define Y second
// #define short unsigned short
#define pii pair<int, int>
#define pb push_back
#define base complex<double>
#define ll long long
#define ull unsigned ll
// #define ui int_fast64_t
// #define double long double
#define fast ios_base::sync_with_stdio(0); cin.tie(0); cout.tie(0)

using namespace std;
using namespace __gnu_cxx;

template<typename T, typename G>
bool rmax(T &a, G b1) {
    T b = static_cast<T>(b1);
    if (a < b) {
        a = b;
        return 1;
    }
    return 0;
}

template<typename T, typename G>
bool rmin(T &a, G b) {
    T b1 = static_cast<T>(b);
    if (b1 < a) {
        a = b1;
        return 1;
    }
    return 0;
}

template<typename T>
istream &operator>>(istream &in, vector<T> &a) {
```

```

    for (auto &it : a) {
        in >> it;
    }
    return in;
}

template<typename T>
ostream &operator<<(ostream &out, vector<T> &a) {
    for (auto &it : a) {
        out << it << ' ';
    }
    return out;
}

int mex(vector<int> &a) {
    for (int i = 0; i <= a.size(); i++) {
        bool have = 0;
        for (auto &it : a) {
            if (it == i) {
                have = 1;
                break;
            }
        }
        if (!have) {
            return i;
        }
    }
}

vector<int> func_grandi(int b) {
    vector<int> my(b * 2 + 2);
    my[0] = 0;

    for (int i = 1; i <= b; i++) {
        my[i] = i;
    }

    vector<int> all_of;
    all_of.emplace_back(my[b]);
    my[b + 1] = mex(all_of);
    all_of.emplace_back(my[b + 1]);

    // cout << all_of << endl;
    for (int b1 = b + 2; b1 < my.size(); b1++) {
        my[b1] = mex(all_of);
        all_of.emplace_back(my[b1]);
    }
    return my;
}

pii make_choice(pii a) {
    cout << a.X << ' ' << a.Y << endl;
    if (a.X == -1 && a.Y == -1) {
        exit(0);
    }
    cin >> a.X >> a.Y;

    return a;
}

void solve() {
    int n;
    cin >> n;

    vector<int> f;
    vector<int> a1(n);
    cin >> a1;

    f = a1;

    vector<vector<int>> b1;
    for (int i = 0; i < n; i++) {
        b1.emplace_back(func_grandi(a1[i]));
    }
}

```

```

int xr = 0;
for (auto &it : b1) {
    xr ^= it.back();
}

for (int i = 0; i < n; i++) {
    a1[i] += a1[i] + 1;
}

if (xr == 0) {
    cout << -1 << '␣' << -1 << endl;
    return;
}

int a, b;

auto make_hhod = [&] () -> pii {
    xr = 0;
    for (int i = 0; i < n; i++) {
        xr ^= b1[i][a1[i]];
    }

    for (int i = 0; i < n; i++) {
        if (a1[i] != 0) {
            xr ^= b1[i][a1[i]];

            if (a1[i] == f[i] + 1) {
                xr ^= b1[i][f[i]];
                if (xr == 0) {
                    a1[i] = f[i];
                    return {i + 1, 0};
                }
                xr ^= b1[i][f[i]];
                continue;
            }

            if (a1[i] > f[i] + 1) {
                xr ^= b1[i][f[i]];

                if (xr == 0) {
                    a1[i] = f[i];
                    return {i + 1, 0};
                }
                xr ^= b1[i][f[i]];

                for (int prev = f[i] + 1; prev < a1[i]; prev++) {
                    xr ^= b1[i][prev];

                    if (xr == 0) {
                        int diff = a1[i] - prev;
                        a1[i] = prev;
                        return {i + 1, diff};
                    }

                    xr ^= b1[i][prev];
                }
            } else {
                for (int prev = 0; prev < a1[i]; prev++) {
                    xr ^= b1[i][prev];

                    if (xr == 0) {
                        int diff = a1[i] - prev;
                        a1[i] = prev;
                        return {i + 1, diff};
                    }

                    xr ^= b1[i][prev];
                }
            }

            xr ^= b1[i][a1[i]];
        }
    }
}

```

```

    }

    return {-1, -1};
};

pii fg = {-1, -1};

while (true) {
    pii g = make_choice(make_hhod());

    if (g == fg) {
        return;
    }

    g.X--;

    if (g.Y == 0) {
        a1[g.X] = f[g.X];
    } else {
        a1[g.X] -= g.Y;
    }
}

}

signed main() {
    // fast;
    // cout.precision(10);
    // cout << fixed;

    // #if defined(LOCAL_CLION)
    //     freopen("../input.txt", "r", stdin);
    //     freopen("../output.txt", "w", stdout);
    // #elif defined(LOCAL)
    //     freopen("../input.txt", "r", stdin);
    //     freopen("../output.txt", "w", stdout);
    // #else
    //     //     freopen("../mining.in", "r", stdin);
    //     //     freopen("../mining.out", "w", stdout);
    // #endif

    int t = 1;
    // cin >> t;

    while (t--)
        solve();
}

//clang-format on

```

Task E ()

```
// Created by a on 06.03.2022.
//
// clang-format off

// #include <random>

#include <algorithm>
#include <numeric>
#include <string>
#include <unordered_set>
#pragma GCC target ("sse, sse2, sse3, ssse3, popcnt, abm, mmx, tune=native")
#pragma GCC optimize ("fast-math")
#pragma GCC optimize ("O3", "no-stack-protector", "inline", "unroll-loops")
#pragma comment(linker, "/stack:200000000")
// #pragma GCC target ("avx, avx2, fma")

#include <bits/stdc++.h>
#include <ext/rope>

// #include <ext/pb_ds/assoc_container.hpp>
// #include <ext/pb_ds/tree_policy.hpp>
// using namespace __gnu_pbds;

// #define ordered_set tree<int, null_type, less<int>, rb_tree_tag,
//     tree_order_statistics_node_update>

#define endl "\n"
// #define int int_fast32_t
#define int long long
#define ui unsigned int
#define all(a) a.begin(), a.end()
#define rall(a) a.rbegin(), a.rend()
#define mall(a) a.mutable_begin(), a.mutable_end()
#define uni(x) sort(all(x)); x.erase(unique(all(x)), x.end())
#define X first
#define Y second
// #define short unsigned short
#define pii pair<int, int>
#define pb push_back
#define base complex<double>
#define ll long long
#define ull unsigned ll
// #define ui int_fast64_t
// #define double long double
#define fast ios_base::sync_with_stdio(0); cin.tie(0); cout.tie(0)

using namespace std;
using namespace __gnu_cxx;

template<typename T, typename G>
bool rmax(T &a, G b1) {
    T b = static_cast<T>(b1);
    if (a < b) {
        a = b;
        return 1;
    }
    return 0;
}

template<typename T, typename G>
bool rmin(T &a, G b) {
    T b1 = static_cast<T>(b);
    if (b1 < a) {
        a = b1;
        return 1;
    }
    return 0;
}

template<typename T>
istream &operator>>(istream &in, vector<T> &a) {
```

```

    for (auto &it : a) {
        in >> it;
    }
    return in;
}

template<typename T>
ostream &operator<<(ostream &out, vector<T> &a) {
    for (auto &it : a) {
        out << it << ' ';
    }
    return out;
}

vector<string> pows;

void solve() {
    int n;
    cin >> n;

    int cnt = 0;
    for (int pow = 9; pow >= 0; pow--) {
        int num = (1 << pow);
        while (n >= num) {
            n -= num;
            // cout << n << endl;
            cout << pows[pow] << endl;
            cnt++;
        }
    }

    string df = "0";
    while (df.size() != 10) {
        df += "0";
    }

    while (cnt != 10) {
        cnt++;
        cout << df << endl;
    }
    cout << endl;
    // return;
}

void solve_recovery() {
    int ans = 0;
    for (int i = 0; i < 10; i++) {
        string f;
        cin >> f;
        int pow = -1;
        for (char &it : f) {
            if (it == '1') {
                pow++;
            }
        }
        if (pow == -1) {
            continue;
        }
        ans += (1 << (pow));
    }

    cout << ans << endl;
}

signed main() {
    fast;
    // cout.precision(10);
    // cout << fixed;

#ifdef LOCAL_CLION
    freopen("../input.txt", "r", stdin);
    freopen("../output.txt", "w", stdout);
#endif
}

```

```

#elif defined(LOCAL)
    freopen("./input.txt", "r", stdin);
    freopen("./output.txt", "w", stdout);
#else
    //    freopen("./mining.in", "r", stdin);
    //    freopen("./mining.out", "w", stdout);
#endif

    pows.resize(10);
    string fg = "0";
    while (fg.size() != 10) {
        fg += "0";
    }

    for (int i = 0; i < 10; i++) {
        fg[i] = '1';
        pows[i] = fg;
    }

    int t = 1;
    cin >> t;
    // string fg;
    cin >> fg;

    while (t--) {
        if (fg == "transmit") {
            solve();
        } else {
            solve_recovery();
        }
    }
}

//clang-format on

```


Task F ()

```
// Created by a on 06.03.2022.
//
// clang-format off

// #include <random>

#include <algorithm>
#include <numeric>
#include <unordered_set>
#pragma GCC target("sse,sse2,sse3,ssse3,popcnt,abm,mmx,tune=native")
#pragma GCC optimize("fast-math")
#pragma GCC optimize("O3", "no-stack-protector", "inline", "unroll-loops")
#pragma comment(linker, "/stack:200000000")
// #pragma GCC target("avx,avx2,fma")

#include <bits/stdc++.h>
#include <ext/rope>

// #include <ext/pb_ds/assoc_container.hpp>
// #include <ext/pb_ds/tree_policy.hpp>
// using namespace __gnu_pbds;

// #define ordered_set tree<int, null_type, less<int>, rb_tree_tag,
//     tree_order_statistics_node_update>

#define endl "\n"
// #define int int_fast32_t
#define int long long
#define ui unsigned int
#define all(a) a.begin(), a.end()
#define rall(a) a.rbegin(), a.rend()
#define mall(a) a.mutable_begin(), a.mutable_end()
#define uni(x) sort(all(x)); x.erase(unique(all(x)), x.end())
#define X first
#define Y second
// #define short unsigned short
#define pii pair<int, int>
#define pb push_back
#define base complex<double>
#define ll long long
#define ull unsigned ll
// #define ui int_fast64_t
// #define double long double
#define fast ios_base::sync_with_stdio(0); cin.tie(0); cout.tie(0)

using namespace std;
using namespace __gnu_cxx;

template<typename T, typename G>
bool rmax(T &a, G b1) {
    T b = static_cast<T>(b1);
    if (a < b) {
        a = b;
        return 1;
    }
    return 0;
}

template<typename T, typename G>
bool rmin(T &a, G b) {
    T b1 = static_cast<T>(b);
    if (b1 < a) {
        a = b1;
        return 1;
    }
    return 0;
}

template<typename T>
istream &operator>>(istream &in, vector<T> &a) {
    for (auto &it : a) {
```

```

        in >> it;
    }
    return in;
}

template<typename T>
ostream &operator<<(ostream &out, vector<T> &a) {
    for (auto &it : a) {
        out << it << ' ';
    }
    return out;
}

using number = vector<pii>;

void normal(number &a, number &b) {
    number new_a, new_b;

    while (a.size() > 0 || b.size() > 0) {
        if (a.size() == 0) {
            while (b.size() > 0) {
                new_a.push_back({0, b.back().Y});
                new_b.push_back(b.back());
                b.pop_back();
            }
            break;
        }

        if (b.size() == 0) {
            while (a.size() > 0) {
                new_b.push_back({0, a.back().Y});
                new_a.push_back(a.back());
                a.pop_back();
            }
            break;
        }

        pii it = a.back();
        pii new_it = b.back();

        if (it.Y < new_it.Y) {
            new_a.push_back(it);
            new_b.push_back({new_it.X, it.Y});
            a.pop_back();
            b.back().Y -= it.Y;
        } else if (it.Y > new_it.Y) {
            new_b.push_back(new_it);
            new_a.push_back({it.X, new_it.Y});
            a.back().Y -= new_it.Y;
            b.pop_back();
        } else {
            new_b.push_back(new_it);
            new_a.push_back(it);
            a.pop_back();
            b.pop_back();
        }
    }

    number my;

    int carry = 0;
    for (int i = 0; i < new_a.size(); i++) {
        if (new_a[i].X + new_b[i].X >= 10) {
            if (carry == 0) {
                int val = new_a[i].X + new_b[i].X;
                my.push_back({val % 10, 1});
                my.push_back({(val + 1) % 10, new_b[i].Y - 1});
                carry = 1;
            } else {
                int val = new_a[i].X + new_b[i].X + carry;
                my.push_back({val % 10, new_b[i].Y});
                carry = 1;
            }
        }
    }
}

```

```

    } else {
        if (carry == 0) {
            my.push_back({new_a[i].X + new_b[i].X, new_a[i].Y});
            carry = 0;
        } else {
            int val = new_a[i].X + new_b[i].X + carry;
            if (val == 10) {
                my.push_back({0, new_a[i].Y});
                carry = 1;
            } else {
                my.push_back({val, 1});
                my.push_back({val - 1, new_a[i].Y - 1});
                carry = 0;
            }
        }
    }
}

if (carry != 0)
    my.push_back({carry, 1});
reverse(all(my));

for (auto &it : my) {
    if (it.Y == 0) {
        continue;
    }
    if (it.Y != 1) {
        cout << "(" << it.X << '|' << it.Y << ")";
    } else {
        cout << it.X;
    }
}
cout << endl;
}

number from_string(string &a) {
    number new_a;

    for (int i = 0; i < a.size(); i++) {
        if (a[i] == '(') {
            int num = a[i + 1] - '0';
            i += 2;
            int cnt = 0;
            while (a[i] != ')') {
                if (a[i] == '|') {
                    i++;
                }
                cnt = cnt * 10 + a[i] - '0';
                i++;
            }

            new_a.push_back({num, cnt});
        } else {
            new_a.push_back({a[i] - '0', 1});
        }
    }

    return new_a;
}

void solve() {
    string a, b;
    cin >> a;

    cin >> b;

    number a1 = from_string(a);
    number b1 = from_string(b);

    // for (pii &it : a1) {
    //     cout << it.X << ' ' << it.Y << endl;

```

```

    // }

    normal(a1, b1);
}

signed main() {
    fast;
    // cout.precision(10);
    // cout << fixed;

    #if defined(LOCAL_CLION)
        freopen("../input.txt", "r", stdin);
        freopen("../output.txt", "w", stdout);
    #elif defined(LOCAL)
        freopen("./input.txt", "r", stdin);
        freopen("./output.txt", "w", stdout);
    #else
        // freopen("./mining.in", "r", stdin);
        // freopen("./mining.out", "w", stdout);
    #endif

    int t = 1;
    // cin >> t;

    while (t--)
        solve();
}

//clang-format on

```