

# Олимпиада СПбГУ по информатике 2021/22 учебного года

A	B	C	D	E	F	Sum
100	100	100	60	58	31	449

## Task A ()

```
#include <bits/stdc++.h>
using namespace std;
typedef long long ll;
typedef long double ld;
#define len(a) (int)a.size()
#define all(x) x.begin(), x.end()
#define rall(x) x.rbegin(), x.rend()

void sol() {
    int n;
    cin >> n;
    map<ll, ll> x;
    for (int i = 0; i < n; ++i) {
        int a, b;
        cin >> a >> b;
        int res = b;
        while (a > 0 && a % 10 == 0) {
            res++;
            a /= 10;
        }
        x[res] += a;
    }
    while (1) {
        auto e = x.begin();
        if (e->second % 10 != 0) {
            cout << e->first << endl;
            return;
        } else {
            x[e->first + 1] += e->second / 10;
            x.erase(x.begin());
        }
    }
}

int32_t main() {
    ios::sync_with_stdio(false);
    cin.tie(0);
    cout << fixed << setprecision(30);
    sol();
    return 0;
}
```

## Task B ()

```
#include <bits/stdc++.h>
using namespace std;
typedef long long ll;
typedef long double ld;
#define len(a) (int)a.size()
#define all(x) x.begin(), x.end()
#define rall(x) x.rbegin(), x.rend()

void sol() {
    int n;
    cin >> n;
    vector<int> k(n);
    for (auto &e : k) cin >> e;
    for (int i = n - 1; i >= 0; --i) {
        int t = 0;
        while (t < k[i]) {
            cout << "Flip_and_wait" << endl;
            int cnt = 0;
            string s;
            cin >> s;
            int x = count(all(s), 'e') / 2;
            cnt += x;
            while (cnt < i + 1) {
                cout << "Wait" << endl;
                cin >> s;
                x = count(all(s), 'e') / 2;
                cnt += x;
            }
            t++;
        }
    }
    cout << "Stop" << endl;
}

int32_t main() {
    ios::sync_with_stdio(false);
    cin.tie(0);
    cout << fixed << setprecision(30);
    sol();
    return 0;
}
```

## Task C ()

```
#include <bits/stdc++.h>
using namespace std;
typedef long long ll;
typedef long double ld;
#define len(a) (int)a.size()
#define all(x) x.begin(), x.end()
#define rall(x) x.rbegin(), x.rend()

void paint(string &s, char c) {
    for (auto &e : s) if (e == c) e = '?';
}

void ce(string s) {
    int cnt0 = count(all(s), '0'), cnt1 = count(all(s), '1');
    if (cnt0 == cnt1) {
        if (s[0] == '0') paint(s, '0');
        else paint(s, '1');
    } else if (cnt0 == 0) {
        for (int i = 1; i < len(s); i += 2) s[i] = '?';
    } else if (cnt1 == 0) {
        for (int i = 1; i < len(s); i += 2) s[i] = '?';
    } else {
        if (cnt0 > cnt1) paint(s, '0');
        else paint(s, '1');
    }
    cout << s << '\n';
}

void co(string s) {
    int cnt0 = count(all(s), '0'), cnt1 = count(all(s), '1');
    if (cnt0 == 0 || cnt1 == 0) {
        for (int i = 0; i < len(s) / 2; ++i) s[i] = '?';
    } else {
        if (cnt0 > cnt1) paint(s, '0');
        else paint(s, '1');
    }
    cout << s << '\n';
}

void c(string s) {
    if (len(s) % 2 == 0) ce(s);
    else co(s);
}

void depaint(string &s, char c) {
    for (auto &e : s) if (e == '?') e = c;
}

void de(string s) {
    int cnt0 = count(all(s), '0'), cnt1 = count(all(s), '1');
    if (cnt0 == len(s) / 2) {
        if (s[0] == '?') depaint(s, '1');
        else depaint(s, '0');
    } else if (cnt1 == len(s) / 2) {
        if (s[0] == '?') depaint(s, '0');
        else depaint(s, '1');
    } else {
        if (cnt0 == 0) depaint(s, '0');
        else depaint(s, '1');
    }
    cout << s << '\n';
}

void dod(string s) {
    int cnt0 = count(all(s), '0'), cnt1 = count(all(s), '1');
    if (cnt0 == len(s) / 2 + 1) depaint(s, '0');
    else if (cnt1 == len(s) / 2 + 1) depaint(s, '1');
    else {
        if (cnt0 == 0) depaint(s, '0');
        else depaint(s, '1');
    }
    cout << s << '\n';
}
```

```

}

void d(string s) {
    if (len(s) % 2 == 0) de(s);
    else dod(s);
}

void sol() {
    int t;
    cin >> t;
    for (;t--;) {
        string s;
        cin >> s;
        if (count(all(s), '?') == 0) c(s);
        else d(s);
    }
}

int32_t main() {
    ios::sync_with_stdio(false);
    cin.tie(0);
    cout << fixed << setprecision(30);
    sol();
    return 0;
}

```

## Task D ()

```
#include <bits/stdc++.h>
using namespace std;
typedef long long ll;
typedef long double ld;
#define len(a) (int)a.size()
#define all(x) x.begin(), x.end()
#define rall(x) x.rbegin(), x.rend()

const ll MOD = 998244353;

void sol() {
    ll n;
    cin >> n;
    n++;
    ll cnt = MOD * (n / MOD) * (n / MOD - 1) / 2 + (n % MOD + 1) * (n / MOD);
    cout << cnt << '\n';
    auto f = [&](ll x) {
        ll res = 1ll;
        for (int i = 2; i <= x; ++i) res = res * i % MOD;
        return res;
    };
    auto pm = [&](ll x, ll y) {
        ll res = 1;
        for (; y > 0; y >>= 1) {
            if (y & 1) res = res * x % MOD;
            x = x * x % MOD;
        }
        return res % MOD;
    };
    ll res = f(n % (MOD - 2));
    cout << res << endl;
}

int32_t main() {
    ios::sync_with_stdio(false);
    cin.tie(0);
    cout << fixed << setprecision(30);
    sol();
    return 0;
}
```

## Task E ()

```
#include <bits/stdc++.h>
using namespace std;
typedef long long ll;
typedef long double ld;
#define len(a) (int)a.size()
#define all(x) x.begin(), x.end()
#define rall(x) x.rbegin(), x.rend()

struct t {
    int a, b, d;
    t() {}
    t(int a, int b, int d): a(a), b(b), d(d) {}
};

void sol() {
    int n;
    cin >> n;
    vector<int> v(n);
    for (auto &e : v) cin >> e;
    int q;
    cin >> q;
    vector<t> req(q);
    for (int i = 0; i < q; ++i) {
        int a, b, d;
        cin >> a >> b >> d;
        req[i] = t(a, b, d);
    }
    for (auto e : req) {
        int res = 0;
        for (int i = e.a; i < e.b; ++i) {
            res += (e.d + v[i] - 1) / v[i];
        }
        cout << res << endl;
    }
}

int32_t main() {
    ios::sync_with_stdio(false);
    cin.tie(0);
    cout << fixed << setprecision(30);
    sol();
    return 0;
}
```

## Task F ()

```
#include <bits/stdc++.h>
using namespace std;
typedef long long ll;
typedef long double ld;
#define len(a) (int)a.size()
#define all(x) x.begin(), x.end()
#define rall(x) x.rbegin(), x.rend()

void sol() {
    int n;
    cin >> n;
    vector<int> l(n), r(n);
    for (int i = 0; i < n; ++i) {
        cin >> l[i] >> r[i];
    }
    vector<ll> pl(n + 1), pr(n + 1);
    for (int i = 0; i < n; ++i) {
        pr[i + 1] = pr[i] + r[i];
        pl[i + 1] = pl[i] + l[i];
    }

    vector<int> dp(n + 1);
    for (int i = 1; i <= n; ++i) {
        for (int j = 0; j < i; ++j) {
            if (pr[i] - pr[j] >= 0 && pl[i] - pl[j] <= 0) dp[i] = max(dp[i], dp[j] + 1);
        }
    }
    cout << *max_element(all(dp)) + 1 << endl;
}

int32_t main() {
    ios::sync_with_stdio(false);
    cin.tie(0);
    cout << fixed << setprecision(30);
    sol();
    return 0;
}
```