

# Олимпиада СПбГУ по информатике 2021/22 учебного года

A	B	C	D	E	F	Sum
100	100	100	80	58	0	438

## Task A ()

```
#include <bits/stdc++.h>
#define f first
#define s second
using namespace std;

signed main() {
    int n; cin >> n;
    vector<pair<int, int>> v(n);
    for (auto &x : v) {
        cin >> x.first >> x.second;
    }

    map<int, int> mp;
    for (auto x : v) {
        int nm = 1;
        while (x.first > 0) {
            mp[x.second + nm] += x.first % 10;
            x.first /= 10;
            ++nm;
        }
    }

    for (auto &p : mp) {
        if (p.second % 10) {
            cout << p.first - 1;
            return 0;
        }
        mp[p.first + 1] += p.second / 10;
    }
}
```

## Task B ()

```
#include <bits/stdc++.h>

using namespace std;

int get() {
    string bp;
    cin >> bp;

    if (bp == "Fail" || bp == "Tired" || bp == "Burn") exit(0);

    cout << endl;
    return (bp.size() - 3) / 2;
}

void wait() {
    cout << "Wait";
    cout << endl;
}

void flip() {
    cout << "Flip_and_wait" << endl;
}

void stop() {
    cout << "Stop" << endl;
    exit(0);
}

signed main() {
    int n; cin >> n;
    vector<int> v(n);
    for (auto &x : v) cin >> x;
    int last = 1e9 + 57;
    for (int i = 0; i < n; ++i) {
        if (v[i]) {
            last = i;
            break;
        }
    }

    flip();
    for (int i = n - 1; i > -1; --i) {
        while (v[i]--) {
            int sum = 0;
            while (sum < i + 1) {
                sum += get();
                if (sum < i + 1) wait();
            }
            if (!(i == last && v[i] == 0)) flip();
        }
    }
    stop();
}
```

## Task C ()

```
#include <bits/stdc++.h>

using namespace std;

void code(string s) {
    int n = s.size();

    int nm0 = 0, nm1 = 0;
    for (int i = 0; i < n; ++i) {
        nm0 += (s[i] == '0');
        nm1 += (s[i] == '1');
    }

    if (nm0 >= nm1) { // n / 2
        int num = 0;
        for (auto &x : s) {
            if (num < n / 2 && x == '0') { x = '?'; num++; }
        }
    }
    if (nm1 > nm0) { // n / 2 + 1
        int num = 0;
        for (auto &x : s) {
            if (num < n / 2 + 1 && x == '1') { x = '?'; num++; }
        }
    }
    cout << s << '\n';
}

void decode(string s) {
    int n = s.size();
    int nmV = 0;
    for (auto &x : s) nmV += (x == '?');

    char c = char('0' + nmV - n / 2);

    for (auto &x : s) x = (x == '?' ? c : x);

    cout << s << '\n';
}

void solve() {
    string s; cin >> s;
    bool c = 0;
    for (auto &x : s) if (x == '?') c = 1;
    if (c) decode(s);
    if (!c) code(s);
}

signed main() {
    int n; cin >> n;
    while (n--) {
        solve();
    }
}
```

## Task D ()

```
#include <bits/stdc++.h>
#define int long long
using namespace std;

const int mod = 998244353;

int sm(int a, int b) {
    return (a + b) % mod;
}

int mlt(int a, int b) {
    return (a * b) % mod;
}

pair<int, int> solve(int n) {
    vector<int> pr(3);
    pr[0] = 0;
    pr[1] = 1;
    for (int i = 2; i <= n; ++i) {
        int x = mlt(sm(pr[0], 1), mlt(i, i));
        pr[2] = sm(pr[1], x);

        pr[0] = pr[1];
        pr[1] = pr[2];
    }

    return { sm(pr[0], 1), sm(pr[1], 1) };
}

vector<pair<int, int>> precalc(11);

pair<int, int> solve_big(int n) {
    vector<int> pr(3);
    pr[0] = max(0ll, precalc[(int)(n / 1e8)].first - 1);
    pr[1] = max(0ll, precalc[(int)(n / 1e8)].second - 1);
    for (int i = ((int)(n / 1e8)) * 1e8 + 1; i <= n; ++i) {
        int x = mlt(sm(pr[0], 1), mlt(i, i));
        pr[2] = sm(pr[1], x);

        pr[0] = pr[1];
        pr[1] = pr[2];
    }

    return { sm(pr[0], 1), sm(pr[1], 1) };
}

signed main() {
    ios_base::sync_with_stdio(0);
    cin.tie(0); cout.tie(0);
    precalc[1] = solve(1e8);

    precalc = { { 0, 0 },
        { 808258749, 980201571 },
        { 117153405, 468660588 },
        { 761699708, 478389955 },
        { 573994984, 137136996 },
        { 62402409, 203730522 },
        { 511621808, 533107751 },
        { 242726978, 496409701 },
        { 887890124, 637333574 },
        { 875880304, 465585441 },
        { 0, 0 }
    };

    int n; cin >> n;

    if (n < 1e8) {
        cout << 0 << '␣' << solve(n).second << '\n';
    } else {
        cout << 0 << '␣' << solve_big(n).second << '\n';
    }
}
```

```

}

/*

void solve_big(int n) {

    vector<int> pr(3);
    pr[0] = precalc[(int)(n / 1e8)].first;
    pr[1] = precalc[(int)(n / 1e8)].second;
    for (int i = ((int)(n / 1e8)) * 1e8 + 1; i <= n; ++i) {
        int x = mlt(sm(pr[0], 1), mlt(i, i));
        pr[2] = sm(pr[1], x);

        pr[0] = pr[1];
        pr[1] = pr[2];
    }

    return { sm(pr[0], 1), sm(pr[1], 1) };
}

*/

```

## Task E ()

```
#include <bits/stdc++.h>

using namespace std;

signed main() {
    ios_base::sync_with_stdio(0);
    cin.tie(0); cout.tie(0);
    int n; cin >> n;
    vector<int> v(n);
    for (auto &x : v) cin >> x;
    int q; cin >> q;
    while (q--) {
        int a, b, d;
        cin >> a >> b >> d;
        int res = 0;
        for (int i = b - 1; i >= a; --i) {
            res += (d + v[i] - 1) / v[i];
        }
        cout << res << '\n';
    }
}
```

## Task F ()

```
#include <bits/stdc++.h>
#define int long long
using namespace std;

struct seg {
    int l, r;
    seg() {}
    seg(int a, int b) {
        l = a;
        r = b;
    }
};

seg intersect(seg a, seg b) {
    int lo = 0;

    return {max(a.l, b.l), min(a.r, b.r) };
}

signed main() {
    int n; cin >> n;
    vector<seg> v(n);
    for (auto &x : v) cin >> x.l >> x.r;
    vector<int> dp(n + 1);
    vector<seg> segdp(n + 1);
    vector<int> mn(n + 1);
    vector<int> mx(n + 1);
    dp[0] = 1;
    segdp[0] = {0, 0};
    mn[0] = mx[0] = 0;

    for (int i = 1; i < n + 1; ++i) {
        seg sg = v[i - 1];
        mn[i] = mn[i - 1] + sg.l;
        mx[i] = mx[i - 1] + sg.r;
        dp[i] = 1;
        segdp[i] = { mn[i], mx[i] };

        for (int j = 0; j < i; ++j) {
            seg q = { mn[i] - mn[j] + segdp[j].l, mx[i] - mx[j] + segdp[j].r };
            seg inter = intersect(q, segdp[j]);
            if (inter.r - inter.l >= 0 && dp[j] + 1 >= dp[i]) {
                dp[i] = dp[j] + 1;
                segdp[i] = inter;
            }
        }
    }

    int mxx = 0;
    for (auto &x : dp) { mxx = max(mxx, x); }
    cout << mxx;
}
```