

Олимпиада СПбГУ по информатике 2020/21 учебного года

A	B	C	D	E	F	Sum
100	100	100	100	55	25	480

Task A ()

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

// #pragma GCC optimize("Ofast")

using namespace std;

typedef long long ll;
typedef long double ld;
#define pb push_back
#define pll pair<ll, ll>

const ll maxn = 11(1e5) + 5;
const ll inf = 11(1e12) + 123;

void solve() {
    ll k;
    cin >> k;
    if(k < 10) {
        cout << (k % 10) << endl;
    } else {
        k -= 10;
        if(k % 9 == 0) {
            cout << "0\n";
        } else {
            cout << (k % 9) + 1 << endl;
        }
    }
}

int main()
{
    //freopen("input.txt", "r", stdin);
    // freopen("output.txt", "w", stdout);

    ios::sync_with_stdio(0);
    cin.tie(0);
    cout.tie(0);

    solve();

    return 0;
}
```

Task B ()

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

// #pragma GCC optimize("Ofast")

using namespace std;

typedef long long ll;
typedef long double ld;
#define pb push_back
#define pll pair<ll, ll>

const ll maxn = 11(1e5) + 5;
const ll inf = 11(1e12) + 123;

void solve() {
    ll n, k;
    cin >> n >> k;
    string s;
    cin >> s;

    set<char> lets;
    ll cur_cnt = 0;

    ll ans = 1;
    for (ll i = 0; i < n; i++) {
        if (cur_cnt == k || (lets.size() == 3 && lets.count(s[i]) == 0)) {
            cur_cnt = 0;
            lets.clear();
            ans++;
        }

        lets.insert(s[i]);
        cur_cnt++;
    }

    cout << ans << endl;
}

int main()
{
    //freopen("input.txt", "r", stdin);
    // freopen("output.txt", "w", stdout);

    ios::sync_with_stdio(0);
    cin.tie(0);
    cout.tie(0);

    solve();

    return 0;
}
```

Task C ()

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

// #pragma GCC optimize("Ofast")

using namespace std;

typedef int ll;
typedef long double ld;
#define pb push_back
#define pll pair<ll, ll>

const ll maxn = 11(1e5) + 5;
const ll inf = 11(1e9) + 123;

void solve() {
    ll n, x, y;
    cin >> n >> x >> y;
    vector<ll> vs;
    vector<ll> weigs;
    for (ll i = 0; i < n; i++) {
        ll cur;
        cin >> cur;
        vs.pb(cur);
    }
    for (ll i = 0; i < n; i++) {
        ll cur;
        cin >> cur;
        weigs.pb(cur);
    }

    ll old[x + 1], cur[x + 1];
    vector<ll> par[n];
    for (ll i = 0; i < n; i++) {
        par[i].assign(x + 1, 0);
    }
    for (ll i = 0; i < x + 1; i++) {
        cur[i] = -1;
    }
    cur[x] = 0;

    for (ll i = 0; i < n; i++) {
        for (ll j = 0; j < x + 1; j++) {
            old[j] = cur[j];
            cur[j] = -1;
        }
        for (ll j = 0; j < x + 1; j++) {
            if (old[j] == -1) continue;
            if (j >= vs[i]) {
                if (old[j] + weigs[i] > cur[j - vs[i]]) {
                    cur[j - vs[i]] = old[j] + weigs[i];
                    par[i][j - vs[i]] = j;
                }
            }

            if (old[j] > cur[j]) {
                cur[j] = old[j];
                par[i][j] = j;
            }
        }
    }

    ll mus = 0;
    for (ll i = 0; i < n; i++) {
        mus += weigs[i];
    }

    for (ll i = 0; i < x + 1; i++) {
        if (cur[i] == -1) continue;
        if (mus - cur[i] <= y) {
            /// good
            ll state = i;
        }
    }
}
```

```

        string ans = "";
        for (ll j = n - 1; j >= 0; j--) {
            if (par[j][state] == state) {
                ans += "y";
            } else {
                ans += "x";
            }

            state = par[j][state];
        }
        reverse(ans.begin(), ans.end());
        cout << ans << endl;

        return;
    }
}

cout << "-1\n";
}

int main()
{
    //freopen("input.txt", "r", stdin);
    // freopen("output.txt", "w", stdout);

    ios::sync_with_stdio(0);
    cin.tie(0);
    cout.tie(0);

    solve();

    return 0;
}

```

Task D ()

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

// #pragma GCC optimize("Ofast")

using namespace std;

typedef int ll;
typedef long double ld;
#define pb push_back
#define pll pair<ll, ll>

const ll maxn = 11(1e5) + 5;
const ll inf = 11(1e9) + 123;

void solve() {
    ll n;
    cin >> n;
    string s;
    cin >> s;
    for (ll i = 0; i < 2 * n; i++) {
        if (s[i] == ')') s[i] = '(';
        if (s[i] == ']') s[i] = '[';
    }

    vector<ll> st;
    ll i = 0;
    while (i < 2 * n) {
        ll tp = 0;
        if (s[i] == '[') tp = 1;

        ll j = i;
        while (j < 2 * n && s[j] == s[i]) j++;

        if ((j - i) % 2 == 0) {
            i = j;
        } else {
            if (st.size() > 0 && st.back() == tp) {
                st.pop_back();
            } else {
                st.pb(tp);
            }
            i = j;
        }
    }

    cout << st.size() / 2 << endl;
}

int main()
{
    // freopen("input.txt", "r", stdin);
    // freopen("output.txt", "w", stdout);

    ios::sync_with_stdio(0);
    cin.tie(0);
    cout.tie(0);

    solve();

    return 0;
}
```

Task E ()

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

// #pragma GCC optimize("Ofast")

using namespace std;

typedef int ll;
typedef long double ld;
#define pb push_back
#define pll pair<ll, ll>

const ll maxn = 11(1e5) + 5;
const ll inf = 11(1e9) + 123;

string tp;
vector<ll> fm = {713256};
vector<ll> f100 = {
86977};

vector<ll> ms(1 << 10, 0);
vector<ll> cor(1 << 10, 0);
set<ll> msk;

ll counter = 0;

ll z = 4;

void build(ll val, vector<ll> cur) {
    if(cur.size() == 3) {
        ll cur_msk = 0;
        for(auto el : cur) {
            cur_msk += 1 << el;
        }

        vector<ll> order(10, 0);
        for(ll i = 0; i < 10; i++) {
            order[i] = i;
        }
        random_shuffle(order.begin(), order.end());

        for(auto i : order) {
            if(cur_msk & (1 << i)) continue;
            ll tmp = cur_msk + (1 << i);
            if(msk.count(tmp) == 0) {
                msk.insert(tmp);
                ms[cur_msk] = i + 1;
                cor[tmp] = cur_msk;
                break;
            }
        }
        if(ms[cur_msk] == 0) counter++;
        return;
    }
    if(val == 10) return;

    vector<ll> tmp = cur;
    tmp.pb(val);

    build(val + 1, cur);
    build(val + 1, tmp);
}

ll f(ll val) {
    return (val % f100[0]) + 1;
}

void solve() {
    // srand(2);

    ll n, k;
    cin >> n >> k;
    vector<ll> a;
```

```

set<ll> vals;
for(ll i = 0; i < k; i++) {
    ll cur;
    cin >> cur;
    a.pb(cur);
    vals.insert(cur);
}
if(tp == "clear") {
    ll cur;
    cin >> cur;
    a.pb(cur);
    vals.insert(cur);
}

if(tp == "add") {
    if(n == 1000000) {
        cout << fm[0] << "\n";
    } else if(n == 100000) {
        ll mus = 0;
        for(auto el : a) {
            mus += el;
        }

        ll val = f(mus);
        cout << val << "\n";
    } else { // n == 10
        ll msk = 0;
        for(auto el : a) {
            msk += (1 << (el - 1));
        }

        cout << ms[msk] << "\n";
    }
} else {
    if(n == 1000000) {
        for(auto el : a) {
            if(el != fm[0]) cout << el << "_";
        }
        cout << "\n";
    } else if(n == 100000) {
        ll mus = 0;
        for(auto el : a) {
            mus += el;
        }

        ll val = -1;
        for(ll i = 0; i < k + 1; i++) {
            if(vals.count(f(mus - a[i])) != 0) {
                val = a[i];
            }
        }

        for(auto el : a) {
            if(el != val) {
                cout << el << "_";
            }
        }
        cout << "\n";
    } else {
        ll msk = 0;
        for(auto el : a) {
            msk += (1 << (el - 1));
        }

        ll an = cor[msk];
        for(ll i = 0; i < n; i++) {
            if(an & (1 << i)) {
                cout << i + 1 << "_";
            }
        }
        cout << "\n";
    }
}
}
}

```

```

int main()
{
    //freopen("input.txt", "r", stdin);
    // freopen("output.txt", "w", stdout);

    ios::sync_with_stdio(0);
    cin.tie(0);
    cout.tie(0);

    /*srand(2);
    for (ll i = 0; i < 60; i++) {
        cout << ll(ld(rand()) / 33000 * 100000);
        cout << "," << endl;
    }*/

    build(0, {});
    // cout << counter << endl;

    cin >> tp;
    ll q;
    cin >> q;
    for (ll i = 0; i < q; i++)
        solve();

    return 0;
}

```


Task F ()

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

// #pragma GCC optimize("Ofast")

using namespace std;

typedef int ll;
typedef long double ld;
#define pb push_back
#define pll pair<ll, ll>

const ll maxn = 11(1e5) + 5;
const ll inf = 11(1e9) + 123;

void solve() {
    vector<pll> vs = {{-1, -1}, {-1, 0}, {-1, 1}, {0, -1}, {0, 1}, {1, -1}, {1, 0}, {1, 1}};

    ll n;
    cin >> n;

    if(n <= 8) {
        cout << "4\n0_0\n0_1\n1_1\n1_0\n";
        for(ll i = 0; i < n; i++) {
            cout << vs[i].first << "_" << vs[i].second << "\n";
        }
    }
}

int main()
{
    //freopen("input.txt", "r", stdin);
    // freopen("output.txt", "w", stdout);

    ios::sync_with_stdio(0);
    cin.tie(0);
    cout.tie(0);

    solve();

    return 0;
}
```