

Олимпиада СПбГУ по информатике 2020/21 учебного года

A	B	C	D	E	F	Sum
100	100	100	100	55	0	455

Task A ()

```
#include <iostream>

#define all(x) (x).begin(), (x).end()

using namespace std;

typedef long long ll;

void operator delete(void*);

int main() {
    ll a;
    cin >> a;
    a--;
    ll kek = (a) / 9;
    a %= 9;
    if (kek == 0) cout << a + 1;
    else cout << a + bool(a);
    return 0;
}
```

Task B ()

```
#include <iostream>
#include <string>
#include <set>

#define all(x) (x).begin(), (x).end()

using namespace std;

typedef long long ll;

void operator delete(void*);

ll n, k;

ll answ;

string s;

int main() {
    cin >> n >> k;
    cin >> s;
    set<char> st;
    ll ln = 0;
    for (ll i = 0; i < n; i++) {
        if (st.size() + !st.count(s[i]) > 3 || ln == k) {
            answ++;
            ln = 0;
            st.clear();
        }
        ln++;
        st.insert(s[i]);
    }
    cout << answ + 1;
    return 0;
}
```

Task C ()

```
#include <algorithm>
#include <iostream>
#include <cstring>
#include <string>
#include <bitset>
#include <set>

#define all(x) (x).begin(), (x).end()

using namespace std;

typedef long long ll;

const ll inf = (ll)1e18 + 228;
const ll cst = 1e3 + 5;

ll n, X, Y;

ll v[cst];
ll w[cst];

int dp[505][250005];
bitset<250005> pr[505];

int main() {
    cin >> n >> X >> Y;
    memset(dp, 0x0f, sizeof(dp));
    dp[0][0] = 0;
    for (ll i = 1; i <= n; i++) cin >> v[i];
    for (ll i = 1; i <= n; i++) cin >> w[i];
    for (ll i = 0; i < n; i++) {
        for (ll j = 0; j <= X; j++) {
            if (dp[i][j] > inf) continue;
            if (v[i + 1] + j <= X) {
                if (dp[i + 1][v[i + 1] + j] > dp[i][j]) {
                    dp[i + 1][v[i + 1] + j] = dp[i][j];
                    pr[i + 1][v[i + 1] + j] = 1;
                }
            }
            if (dp[i + 1][j] > dp[i][j] + w[i + 1]) {
                dp[i + 1][j] = dp[i][j] + w[i + 1];
                pr[i + 1][j] = 0;
            }
        }
    }
    ll ansi = -1;
    for (ll j = 0; j <= X; j++) {
        if (dp[n][j] <= Y) {
            ansi = j;
            break;
        }
    }
    if (ansi == -1) {
        cout << -1;
        return 0;
    }
    string answ = "";
    while (n) {
        if (pr[n][ansi] == 0) {
            answ.push_back('y');
        }
        else {
            answ.push_back('x');
            ansi -= v[n];
        }
        n--;
    }
    reverse(all(answ));
    cout << answ;
    return 0;
}
```

Task D ()

```
#include <algorithm>
#include <iostream>
#include <cstring>
#include <string>
#include <bitset>
#include <vector>
#include <set>

#define all(x) (x).begin(), (x).end()

using namespace std;

typedef long long ll;

const ll inf = (ll)1e18 + 228;
const ll cst = 1e3 + 5;

ll n;

string s;

vector<char> vc;

int main() {
    cin.tie(0) -> sync_with_stdio(0);
    cin >> n >> s;
    n *= 2;
    for (ll i = 0; i < n; i++) {
        if (s[i] == '(' || s[i] == ')') s[i] = 'a';
        if (s[i] == '[' || s[i] == ']') s[i] = 'b';
        if (vc.empty()) vc.push_back(s[i]);
        else {
            if (vc.back() == s[i]) vc.pop_back();
            else vc.push_back(s[i]);
        }
    }
    cout << vc.size() / 2;
    return 0;
}

/*

add
1
100000 5
2 4 6 8 1228

*/
```

Task E ()

```
#include <algorithm>
#include <iostream>
#include <cstring>
#include <string>
#include <bitset>
#include <vector>
#include <random>
#include <set>
#include <map>

#define all(x) (x).begin(), (x).end()

using namespace std;

typedef long long ll;

const ll inf = (ll)1e18 + 228;
const ll cst = 1e3 + 5;

string s;

map<vector<ll>, ll> mp;
map<vector<ll>, ll> mp2;

vector<ll> v1;

ll magic = 998244;

ll gcd(ll a, ll b) {
    return b ? gcd(b, a % b) : a;
}

vector<ll> magicalvc;

void bt(ll v, ll rs = 0) {
    if (rs == 3) {
        vector<ll> v2 = v1;
        mp[v1] = -2334329;
        for (ll i = 1; i <= 10; i++) {
            bool is = 0;
            for (ll j : v1) {
                if (j == i) is = 1;
            }
            if (is) continue;
            v2 = v1;
            v2.push_back(i);
            sort(all(v2));
            if (mp2[v2]) continue;
            mp2[v2] = i;
            mp[v1] = i;
            break;
        }
        return;
    }
    for (ll i = v + 1; i <= 10; i++) {
        v1.push_back(i);
        bt(i, rs + 1);
        v1.pop_back();
    }
}

void solve() {
    if (s == "add") {
        ll n, k;
        cin >> n >> k;
        vector<ll> vc;
        map<ll, ll> is;
        for (ll i = 0; i < k; i++) {
            ll x;
            cin >> x;
            vc.push_back(x);
            is[x] = 1;
        }
    }
}
```

```

    }
    if (n == 10 && k == 3){
        sort(all(vc));
        cout << mp[vc];
    }
    else if (n == 1000000) {
        cout << magic;
    }
    else {
        ll gc = 0;
        for (ll i : vc) gc = gcd(i, gc);
        if (gc == 3) cout << "1337";
        else if (gc == 2) cout << "1337";
        else {
            for (ll i = 0; i < magicalvc.size() - 1; i++) {
                if (!is[magicalvc[i]] && !is[magicalvc[i + 1]]) {
                    cout << magicalvc[i];
                    return;
                }
            }
            throw 1;
        }
    }
}
else {
    ll n, k;
    cin >> n >> k;
    vector<ll> vc;
    map<ll, ll> is;
    for (ll i = 0; i <= k; i++) {
        ll x;
        cin >> x;
        vc.push_back(x);
        is[x] = 1;
    }
    if (n == 10 && k == 3) {
        sort(all(vc));
        ll kek = mp2[vc];
        for (ll i : vc) {
            if (i != kek) cout << i << " ";
        }
    }
    else if (n == 1000000) {
        for (ll i : vc) {
            if (i != magic) cout << i << " ";
        }
    }
    else {
        vector<ll> pfgc;
        ll gc = 0;
        for (ll i : vc) {
            gc = gcd(i, gc);
            pfgc.push_back(gc);
        }
        gc = 0;
        for (ll i = k; i >= 0; i--) {
            ll kek = gc;
            if (i > 0) kek = gcd(kek, pfgc[i - 1]);
            if (kek == 2 && vc[i] == 1337) {
                for (ll j : vc) {
                    if (j != vc[i]) cout << j << " ";
                }
                return;
            }
            if (kek == 3 && vc[i] == 1337) {
                for (ll j : vc) {
                    if (j != vc[i]) cout << j << " ";
                }
                return;
            }
            gc = gcd(vc[i], gc);
        }
        ll kek = 0;
        for (ll i = 0; i < magicalvc.size() - 1; i++) {

```

```

        if (is[magicalvc[i]] && !is[magicalvc[i + 1]]) {
            kek = magicalvc[i];
            break;
        }
    }
    if (!kek) throw 1;
    for (ll j : vc) {
        if (j == kek) continue;
        cout << j << "_";
    }
}

}

ll sm(string s) {
    ll ret = 0;
    for (char c : s) ret += c;
    return ret;
}

mt19937 gen((ll)1e9 + 7);

int main() {
    cin.tie(0) -> sync_with_stdio(0);
    bt(0);
    for (ll i = 1; i <= 30; i++) magicalvc.push_back(gen() % 100000 + 1);
    sort(all(magicalvc));
    cin >> s;
    ll t;
    cin >> t;
    while (t--) {
        solve();
        cout << "\n";
    }
    return 0;
}

/*

*/

```

Task F ()