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 <iostream>
#include <string>
#include <random>
#include <vector>
#include <set>
#include <map>
using namespace std;

typedef long long ll;
typedef long double ld;
typedef pair<int, int> pii;
typedef pair<ll, ll> pll;

#define pb push_back
#define ff first
#define ss second

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

/*

*/
signed main() {
    ios_base::sync_with_stdio(false);cin.tie(0);cout.tie(0);
    int test = 1;
    // cin >> test;
    while(test--){
        solve();
    }
}
```

Task B ()

```
#include <iostream>
#include <string>
#include <random>
#include <vector>
#include <set>
#include <map>
using namespace std;

typedef long long ll;
typedef long double ld;
typedef pair<int, int> pii;
typedef pair<ll, ll> pll;

#define pb push_back
#define ff first
#define ss second

void solve(){
    int n, k;
    cin >> n >> k;
    string s;
    cin >> s;
    int ans = 1;
    set<char> t;
    int sz = 0;
    for(auto i: s){
        if((t.count(i) == 0 && t.size() == 3) || sz == k){
            ans++;
            t.clear();
            sz = 1;
            t.insert(i);
        }
        else sz++, t.insert(i);
    }
    cout << ans;
}

/*

*/
signed main() {
    ios_base::sync_with_stdio(false);cin.tie(0);cout.tie(0);
    int test = 1;
    // cin >> test;
    while(test--){
        solve();
    }
}
```

Task C ()

```
#include <iostream>
#include <string>
#include <random>
#include <vector>
#include <set>
#include <map>
#include <algorithm>
using namespace std;

typedef long long ll;
typedef long double ld;
typedef pair<int, int> pii;
typedef pair<ll, ll> pll;

#define pb push_back
#define ff first
#define ss second

void solve(){
    int n, x, y;
    cin >> n >> x >> y;
    vector<int> w(n), v(n);
    for(auto &i: w) cin >> i;
    for(auto &i: v) cin >> i;
    int sz = 250001;
    vector<vector<int>> dp(n, vector<int> (sz, sz));
    vector<vector<bool>> rev(n, vector<bool> (sz));
    dp[0][w[0]] = 0;
    rev[0][w[0]] = true;
    dp[0][0] = v[0];
    rev[0][0] = false;
    for(int i=1;i<n;i++){
        for(int j=0;j<sz;j++){
            if(dp[i][j] > dp[i-1][j] + v[i]){
                dp[i][j] = dp[i-1][j] + v[i];
                rev[i][j] = false;
            }
            if(j + w[i] < sz){
                if(dp[i][j+w[i]] > dp[i-1][j]){
                    dp[i][j+w[i]] = dp[i-1][j];
                    rev[i][j+w[i]] = true;
                }
            }
        }
    }
    /*for(auto i: dp){
        for(auto j: i) cout << j << ' ';
        cout << '\n';
    }*/
    int pos = -1;
    for(int i=0;i<=x;i++){
        if(dp[n-1][i] <= y){
            pos = i;
            break;
        }
    }
    if(pos == -1){
        cout << "-1\n";
        return;
    }
    string ans;
    int t = n-1;
    while(t >= 0){
        //cout << t << ' ' << pos << ' ' << rev[t][pos] << '\n';
        if(rev[t][pos]){
            ans += 'x';
            pos -= w[t];
        }
        else ans += 'y';
        t--;
    }
    reverse(ans.begin(), ans.end());
    cout << ans << '\n';
}

/*

*/
signed main() {
    //ios_base::sync_with_stdio(false);cin.tie(0);cout.tie(0);
    int test = 1;
    // cin >> test;
    while(test--){
        solve();
    }
}
```

Task D ()

```
#include <iostream>
#include <string>
#include <random>
#include <vector>
#include <set>
#include <map>
#include <algorithm>
using namespace std;

typedef long long ll;
typedef long double ld;
typedef pair<int, int> pii;
typedef pair<ll, ll> pll;

#define pb push_back
#define ff first
#define ss second

void solve(){
    int n;
    cin >> n;
    string s;
    cin >> s;
    n = n*2;
    vector<char> a;
    for(auto i: s){
        if(a.empty()) a.pb(i);
        else{
            if((a.back() == '(' || a.back() == ')') && (i == '(' || i == ')')) a.pop_back();
            else if((a.back() == '[' || a.back() == ']') && (i == '[' || i == ']')) a.pop_back();
            else a.pb(i);
        }
    }
    cout << a.size()/2;
}

/*

*/
signed main() {
    ios_base::sync_with_stdio(false);cin.tie(0);cout.tie(0);
    int test = 1;
    // cin >> test;
    while(test--){
        solve();
    }
}
```

Task E ()

```
#include <iostream>
#include <string>
#include <random>
#include <vector>
#include <set>
#include <map>
#include <unordered_map>
#include <algorithm>
#include <cassert>
using namespace std;

typedef long long ll;
typedef long double ld;
typedef pair<int, int> pii;
typedef pair<ll, ll> pll;

#define pb push_back
#define ff first
#define ss second

map<vector<int>, int> graf, farg;
set<vector<int>> use;

mt19937 rng(239228);

class SegmentTree{
public:
    vector<int> t;
    SegmentTree(int n){
        t.resize(4*n);
    }
    void upd(int v, int tl, int tr, int pos){
        if(tl == tr) t[v] = !t[v];
        else{
            int tm = (tl + tr)/2;
            if(pos <= tm) upd(v*2, tl, tm, pos);
            else upd(v*2+1, tm+1, tr, pos);
            t[v] = t[v*2] + t[v*2+1];
        }
    }
    int fnd(int v, int tl, int tr, int k){
        if(tl == tr) return tl;
        int tm = (tl+tr)/2;
        if((tm - tl + 1) - t[v*2] <= k) return fnd(v*2+1, tm+1, tr, k - ((tm - tl + 1) - t[v*2]));
        return fnd(v*2, tl, tm, k);
    }
    int cnt(int v, int tl, int tr, int l, int r){
        if(tl > r || tr < l) return 0;
        if(l <= tl && tr <= r) return t[v];
        int tm = (tl+tr)/2;
        return cnt(v*2, tl, tm, l, r) + cnt(v*2+1, tm+1, tr, l, r);
    }
};

void gen10(){
    for(int i=1;i<=10;i++){
        for(int j=i+1;j<=10;j++){
            for(int k=j+1;k<=10;k++){
                vector<int> s = {i, j, k};
                //for(auto i: s) cout << i << ' ';
                //cout << '\n';
                for(int h=1;h<=10;h++){
                    if(h == i || h == j || h == k) continue;;
                    vector<int> g = {i, j, k, h};
                    sort(g.begin(), g.end());
                    if(!use.count(g)){
                        //for(auto i: g) cout << i << ' ';
                        //cout << '\n';
                        //cout << '\n';
                        graf[s] = h;
                        farg[g] = h;
                        use.insert(g);
                        break;
                    }
                }
            }
        }
    }
}

void solve1(){
    int n, k;
    cin >> n >> k;
    if(n == 10){
        vector<int> a(k);
        for(auto &i : a) cin >> i;
        sort(a.begin(), a.end());
        cout << graf[a] << '\n';
    }
    else if(n == 1000000){
        set<int> a;
        int s = 0;
        for(int i=0;i<k;i++){
            int x;
            cin >> x;
            a.insert(x);
            s += x;
        }
        s = s%1000000 + 1;
        for(int i=0;i<12;i++){
            if(!a.count(s)){
                cout << s << '\n';
                return;
            }
            s = (s+1)%1000000;
        }
    }
    else{
        vector<ll> a(k);
        for(auto &i: a) cin >> i;
        SegmentTree tree(n);
        ll s = 0;
        for(auto i: a) tree.upd(1, 0, n-1, i-1), s += i*i;
        s = s%n + 1;
        int emp = s-1 - tree.cnt(1, 0, n-1, 0, s-2);
        int emp2 = n - emp - k;
        //cout << s << ' ' << emp << ' ' <<  emp2 << '\n';
        int tmp;
        if(emp2 == 0)  tmp = tree.fnd(1, 0, n-1, 0);
        else tmp = tree.fnd(1, 0, n-1, emp);
        cout << tmp+1 << '\n';
    }
}

void solve2(){
    int n, k;
    cin >> n >> k;
    if(n == 10){
        vector<int> a(k+1);
        for(auto &i : a) cin >> i;
        sort(a.begin(), a.end());
        for(auto i: a) if(i != farg[a]) cout << i << '␣';
        cout << '\n';
    }
    else if(n == 1000000){
        vector<int> a(k+1);
        for(auto &i : a) cin >> i;
        for(int i=0;i<=k;i++){
            vector<int> check;
            for(int j=0;j<=k;j++) if(i != j) check.pb(a[j]);
            set<int> b;
            int s = 0;
            for(auto x: check){
                b.insert(x);
                s += x;
            }
            s = s%1000000 + 1;
            for(int f=0;f<12;f++){
                if(!b.count(s)){
                    if(s == a[i]){
                        for(auto g: check) cout << g << '␣';
                        return;
                    }
                    break;
                }
                s = (s+1)%1000000;
            }
        }
    }
    else{
        SegmentTree tree(n);
        vector<ll> a(k+1);
        ll s = 0;
        for(auto &i: a) cin >> i, tree.upd(1, 0, n-1, i-1), s += i*i;
        shuffle(a.begin(), a.end(), rng);
        int tans = -1;
        for(auto i: a){
            tree.upd(1, 0, n-1, i-1);
            ll ts = (s - i*i)%n + 1;
            int emp = ts-1 - tree.cnt(1, 0, n-1, 0, ts-2);
            int emp2 = n - emp - k;
            //cout << s << ' ' << emp << ' ' <<  emp2 << '\n';
            int tmp;
            if(emp2 == 0)  tmp = tree.fnd(1, 0, n-1, 0);
            else tmp = tree.fnd(1, 0, n-1, emp);
            if(tmp + 1 == i){
                tans = i;
                break;
            }
            tree.upd(1, 0, n-1, i-1);
        }
        assert(tans != -1);
        for(auto i: a) if(i != tans) cout << i << '␣';
        cout << '\n';
    }
}

void solve(){
    gen10();
    string s;
    cin >> s;
    if(s == "add"){
        int t;
        cin >> t;
        while(t--){
            solve1();
        }
    }
    else{
        int t;
        cin >> t;
        while(t--){
            solve2();
        }
    }
}

/*
add
1
10 3
1 7 8

clear
1
10 3
1 8 7 2

add
```

```
2
100000 2
99999 100000
100000 2
99998 99999

clear
2
100000 2
99999 100000 1
100000 2
99999 99998 100000

*/
signed main() {
    ios_base::sync_with_stdio(false);cin.tie(0);cout.tie(0);
    int test = 1;
    // cin >> test;
    while(test--){
        solve();
    }
}
```

Task F ()

```
#include <iostream>
#include <string>
#include <random>
#include <vector>
#include <set>
#include <map>
#include <unordered_map>
#include <algorithm>
using namespace std;

typedef long long ll;
typedef long double ld;
typedef pair<int, int> pii;
typedef pair<ll, ll> pll;

#define pb push_back
#define ff first
#define ss second

void solve(){
    int n;
    cin >> n;
    vector<pair<int, int>> figure = {{0, 0}, {0, 1}, {1, 1}, {1, 0}};
    vector<pair<int, int>> vec = {{1, 1}, {1, 0}, {1, -1}, {0, -1}, {-1, -1}, {-1, 0}, {-1, 1},
        {0, 1}};
    cout << figure.size() << '\n';
    for(auto i: figure) cout << i.ff << '␣' << i.ss << '\n';
    for(int i=0;i<n;i++) cout << vec[i].ff << '␣' << vec[i].ss << '\n';
}

/*

*/
signed main() {
    ios_base::sync_with_stdio(false);cin.tie(0);cout.tie(0);
    int test = 1;
    // cin >> test;
    while(test--){
        solve();
    }
}
```