

Олимпиада СПбГУ по информатике 2020/21 учебного года

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

Task A ()

[illegible]

```
void run() {
    int c;
    cin >> c;
    —c;
    if(c == 0)
        cout << 1;
    else
        cout << ds[c % sz(ds)];
}

signed main() {
    // files("input.txt", "output.txt");
    // files("spiral.in", "spiral.out");
    iostream::sync_with_stdio(0);
    cin.tie(0);
    cout.tie(0);
    int t = 1;
    //cin >> t;
    while(t—){
        run();
        cout.flush();
    }
}
```

Task B ()

```
///#pragma GCC target ("avx2")
///#pragma GCC optimize("O3")
///#pragma GCC optimize("unroll-loops")
///#pragma GCC optimize("Ofast")
#include <bits/stdc++.h>
///#include <ext/rope>
///#include <ext/pb_ds/assoc_container.hpp>
///#include <ext/pb_ds/tree_policy.hpp>

using namespace std;
///using namespace __gnu_cxx;
///using namespace __gnu_pbds;

#define files(fin, fout) freopen(fin, "r", stdin); freopen(fout, "w", stdout)
#define foru(i, n) for(int i = 0; i < n; ++i)
#define fori(i, j, n) for(int i = j; i < n; ++i)
#define vi vector <int>
#define pii pair <int, int>
#define pb push_back
#define mp make_pair
#define mpm(a, b) make_pair(min(a, b), max(a, b))
#define mpint(a, b) ((ll)(a) * (INT_MAX) + (b))
#define all(a) (a).begin(), (a).end()
#define rall(a) (a).rbegin(), (a).rend()
#define mbort(a) sort(all(a))
#define cinm(a) for(auto &i : a) cin >> i
#define cinmm(a) for(auto &i : a) cinm(i)
#define sz(a) (int)(a).size()
#define coutm(a) for(auto &i : a) cout << i << ' '; cout << '\n'
#define coutmm(a) for(auto &i : a) coutm(i)
#define debug(x) cout << #x << ": " << x << '\n'
#define debugm(x) cout << #x << ":\n"; coutm(x)
#define ff first
#define ss second

typedef long long ll;
typedef unsigned long long ull;
typedef long double ld;

const signed inf = 1000000100;
const ll inf64 = 1ll * inf * inf;
const ll mod = 1e9 + 7;

mt19937 rng(0);

#define int long long

const int N = 1e5 + 100;

int n, k;
string s;
vector <int> last(256, -1);
set <int> ones;

struct Node
{
    int x, mn;
    Node(){
        x = inf64;
        mn = inf64;
    }
};

Node t[4 * N];

void update(int v, int tl, int tr, int i, int x){
    if(tl > tr || i < tl || i > tr)
        return;
    if(tl == tr){
        t[v].x = x;
        t[v].mn = x;
        return;
    }
```

```

    }
    int tm = (tl + tr) / 2;
    if(i <= tm)
        update(v * 2 + 1, tl, tm, i, x);
    else
        update(v * 2 + 2, tm + 1, tr, i, x);
    t[v].mn = min(t[2 * v + 1].mn, t[2 * v + 2].mn);
}

int get(int v, int tl, int tr, int l, int r){
    if(tl > tr || l > r || l < tl || r > tr)
        return inf64;
    if(tl == l && tr == r){
        return t[v].mn;
    }
    int tm = (tl + tr) / 2;
    return min(get(2 * v + 1, tl, tm, l, min(r, tm)), get(2 * v + 2, tm + 1, tr, max(l, tm + 1), r));
}

void run() {
    cin >> n >> k;
    cin >> s;
    vector<int> dp(n + 1);
    dp[0] = 0;
    update(0, 0, N - 1, 0, 0);
    for(int i = 0; i < n; ++i){
        if(last[s[i]] == -1){
            last[s[i]] = i + 1;
            ones.insert(i + 1);
        }
        else{
            ones.erase(last[s[i]]);
            last[s[i]] = i + 1;
            ones.insert(i + 1);
        }
        int ti;
        if(ones.size() <= 3){
            ti = 0;
        }
        else{
            auto it = --ones.end();
            --it; // last
            --it; // prev last
            --it; // prev prev last
            ti = (*it);
        }
        ti = max(ti, i - k + 1);
        dp[i + 1] = get(0, 0, N - 1, ti, i) + 1;
        update(0, 0, N - 1, i + 1, dp[i + 1]);
    }
    cout << dp[n];
}

signed main(){
    // files("input.txt", "output.txt");
    // files("spiral.in", "spiral.out");
    iosstream::sync_with_stdio(0);
    cin.tie(0);
    cout.tie(0);
    int t = 1;
    //cin >> t;
    while(t--){
        run();
        cout.flush();
    }
}

```

Task C ()

```
##pragma GCC target ("avx2")
##pragma GCC optimize("O3")
##pragma GCC optimize("unroll-loops")
##pragma GCC optimize("Ofast")
#include <bits/stdc++.h>
#include <ext/rope>
#include <ext/pb_ds/assoc_container.hpp>
#include <ext/pb_ds/tree_policy.hpp>

using namespace std;
using namespace __gnu_cxx;
using namespace __gnu_pbds;

#define files(fin, fout) freopen(fin, "r", stdin); freopen(fout, "w", stdout)
#define foru(i, n) for(int i = 0; i < n; ++i)
#define fori(i, j, n) for(int i = j; i < n; ++i)
#define vi vector<int>
#define pii pair<int, int>
#define pb push_back
#define mp make_pair
#define mpm(a, b) make_pair(min(a, b), max(a, b))
#define mpint(a, b) ((ll)(a) * (INT_MAX) + (b))
#define all(a) (a).begin(), (a).end()
#define rall(a) (a).rbegin(), (a).rend()
#define mbort(a) sort(all(a))
#define cinm(a) for(auto &i : a) cin >> i
#define cinmm(a) for(auto &i : a) cinm(i)
#define sz(a) (int)(a).size()
#define coutm(a) for(auto &i : a) cout << i << ' '; cout << '\n'
#define coutmm(a) for(auto &i : a) coutm(i)
#define debug(x) cout << #x << ": " << x << '\n'
#define debugm(x) cout << #x << ": \n"; coutm(x)
#define ff first
#define ss second

typedef long long ll;
typedef unsigned long long ull;
typedef long double ld;

const signed inf = 1000000100;
const ll inf64 = 1ll * inf * inf;
const ll mod = 1e9 + 7;

mt19937 rng(0);

##define int long long

void run() {
    int n, x, y;
    cin >> n >> x >> y;
    vector<int> v(n);
    cinm(v);
    vector<int> w(n);
    cinm(w);
    vector<int> pdp(x + 100, -inf);
    vector<int> dp(x + 100, -inf);
    vector<vector<int>> prev(n + 1, vector<int>(x + 100, -1));
    dp[x] = y;
    for(int i = 0; i < n; ++i){
        pdp = dp;
        dp = vector<int>(x + 100, -inf);
        for(int px = 0; px <= x; ++px){
            if(px >= v[i]){
                if(pdp[px] > dp[px - v[i]]){
                    dp[px - v[i]] = pdp[px];
                    prev[i + 1][px - v[i]] = px;
                }
            }
            if(pdp[px] >= w[i]){
                if(pdp[px] - w[i] > dp[px]){
                    dp[px] = pdp[px] - w[i];
                    prev[i + 1][px] = px;
                }
            }
        }
    }
}
```

```

        }
    }
}
int last_x = -1;
for (int i = 0; i <= x; ++i){
    if(dp[i] >= 0){
        last_x = i;
        break;
    }
}
if(last_x == -1){
    cout << -1;
    return;
}
string ans;
int tn = n;
while(tn != 0){
    if(prev[tn][last_x] == last_x){
        ans.pb('y');
    }
    else{
        ans.pb('x');
        last_x = prev[tn][last_x];
    }
    tn--;
}
reverse(all(ans));
cout << ans;
}

signed main(){
    //files("input.txt", "output.txt");
    //files("spiral.in", "spiral.out");
    iosstream::sync_with_stdio(0);
    cin.tie(0);
    cout.tie(0);
    int t = 1;
    //cin >> t;
    while(t--){
        run();
        cout.flush();
    }
}

```

Task D ()

```
##pragma GCC target ("avx2")
#pragma GCC optimize("O3")
#pragma GCC optimize("unroll-loops")
##pragma GCC optimize("Ofast")
#include <bits/stdc++.h>
#include <ext/rope>
##include <ext/pb_ds/assoc_container.hpp>
##include <ext/pb_ds/tree_policy.hpp>

using namespace std;
using namespace __gnu_cxx;
//using namespace __gnu_pbds;

#define files(fin, fout) freopen(fin, "r", stdin); freopen(fout, "w", stdout)
#define foru(i, n) for(int i = 0; i < n; ++i)
#define fori(i, j, n) for(int i = j; i < n; ++i)
#define vi vector<int>
#define pii pair<int, int>
#define pb push_back
#define mp make_pair
#define mpm(a, b) make_pair(min(a, b), max(a, b))
#define mpint(a, b) ((ll)(a) * (INT_MAX) + (b))
#define all(a) (a).begin(), (a).end()
#define rall(a) (a).rbegin(), (a).rend()
#define mbort(a) sort(all(a))
#define cinm(a) for(auto &i : a) cin >> i
#define cinmm(a) for(auto &i : a) cinm(i)
#define sz(a) (int)(a).size()
#define coutm(a) for(auto &i : a) cout << i << ' '; cout << '\n'
#define coutmm(a) for(auto &i : a) coutm(i)
#define debug(x) cout << #x << ": " << x << '\n'
#define debugm(x) cout << #x << ":\n"; coutm(x)
#define ff first
#define ss second

typedef long long ll;
typedef unsigned long long ull;
typedef long double ld;

const signed inf = 1000000100;
const ll inf64 = 1ll * inf * inf;
const ll mod = 1e9 + 7;

mt19937 rng(0);

##define int long long

int cost(char from, char to){
    if(from == to)
        return 0;
    if(from == ']''){
        if(to == '[')
            return 0;
        return 1;
    }
    else if(from == '['){
        if(to == ']')
            return 0;
        return 1;
    }
    else if(from == ')'){
        if(to == '(')
            return 0;
        return 1;
    }
    else{
        if(to == ')')
            return 0;
        return 1;
    }
}
```

```

int c(char t){
    if(t == '(' || t == ')')
        return 0;
    return 1;
}

char d(int t){
    if(t == 0)
        return '(';
    else if(t == 1)
        return ')';
    else if(t == 2)
        return '[';
    else
        return ']';
}

void run() {
    int n;
    cin >> n;
    string s;
    cin >> s;
    vector<int> a(2 * n);
    vector<char> paired(2 * n);
    vector<int> id0, id1;
    for(int i = 0; i < 2 * n; ++i){
        a[i] = c(s[i]);
        if(a[i])
            id1.pb(i);
        else
            id0.pb(i);
    }
    int ans = 0;
    rope<int> t;
    for(int i = 0; i < 2 * n; ++i){
        t.push_back(a[i]);
    }
    bool found = 1;
    while(found){
        found = 0;
        /*for(int i = 0; i < sz(t); ++i){
            cout << t[i];
        }
        cout << endl;*/
        for(int i = 0; i < sz(t) - 1; ++i){
            if(t[i] == t[i + 1]){
                t.erase(i, 1);
                t.erase(i, 1);
                found = 1;
            }
        }
    }
    cout << sz(t) / 2;
}

signed main(){
    //files("input.txt", "output.txt");
    //files("spiral.in", "spiral.out");
    iosstream::sync_with_stdio(0);
    cin.tie(0);
    cout.tie(0);
    int t = 1;
    //cin >> t;
    while(t--){
        run();
        cout.flush();
    }
}

```


Task E ()

```
///#pragma GCC target ("avx2")
///#pragma GCC optimize("O3")
///#pragma GCC optimize("unroll-loops")
///#pragma GCC optimize("Ofast")
#include <bits/stdc++.h>
///#include <ext/rope>
///#include <ext/pb_ds/assoc_container.hpp>
///#include <ext/pb_ds/tree_policy.hpp>

using namespace std;
///using namespace __gnu_cxx;
///using namespace __gnu_pbds;

#define files(fin, fout) freopen(fin, "r", stdin); freopen(fout, "w", stdout)
#define foru(i, n) for(int i = 0; i < n; ++i)
#define fori(i, j, n) for(int i = j; i < n; ++i)
#define vi vector<int>
#define pii pair<int, int>
#define pb push_back
#define mp make_pair
#define mpm(a, b) make_pair(min(a, b), max(a, b))
#define mpint(a, b) ((ll)(a) * (INT_MAX) + (b))
#define all(a) (a).begin(), (a).end()
#define rall(a) (a).rbegin(), (a).rend()
#define mbort(a) sort(all(a))
#define cinm(a) for(auto &i : a) cin >> i
#define cinmm(a) for(auto &i : a) cinm(i)
#define sz(a) (int)(a).size()
#define coutm(a) for(auto &i : a) cout << i << ' '; cout << '\n'
#define coutmm(a) for(auto &i : a) coutm(i)
#define debug(x) cout << #x << ": " << x << '\n'
#define debugm(x) cout << #x << ":\n"; coutm(x)
#define ff first
#define ss second

typedef long long ll;
typedef unsigned long long ull;
typedef long double ld;

const signed inf = 1000000100;
const ll inf64 = 1ll * inf * inf;
const ll mod = 1e9 + 7;

mt19937 rng(0);

///#define int long long

const int N = 6;

map<vector<int>, int> precalc;
map<vector<int>, int> precalc1;
map<vector<int>, int> precalc2;
map<vector<int>, int> precalc3;

void build_precalc(){
    precalc[{1,1,1}]=6;
    precalc[{1,1,2}]=6;
    precalc[{1,1,3}]=5;
    precalc[{1,1,4}]=5;
    precalc[{1,1,5}]=9;
    precalc[{1,1,6}]=5;
    precalc[{1,1,7}]=2;
    precalc[{1,1,8}]=6;
    precalc[{1,1,9}]=7;
    precalc[{1,1,10}]=3;
    precalc[{1,2,2}]=9;
    precalc[{1,2,3}]=4;
    precalc[{1,2,4}]=6;
    precalc[{1,2,5}]=6;
    precalc[{1,2,6}]=3;
    precalc[{1,2,7}]=3;
    precalc[{1,2,8}]=6;
```

```

precalc [{1,2,9}]=3;
precalc [{1,2,10}]=8;
precalc [{1,3,3}]=9;
precalc [{1,3,4}]=8;
precalc [{1,3,5}]=10;
precalc [{1,3,6}]=9;
precalc [{1,3,7}]=9;
precalc [{1,3,8}]=9;
precalc [{1,3,9}]=5;
precalc [{1,3,10}]=4;
precalc [{1,4,4}]=2;
precalc [{1,4,5}]=3;
precalc [{1,4,6}]=3;
precalc [{1,4,7}]=10;
precalc [{1,4,8}]=2;
precalc [{1,4,9}]=5;
precalc [{1,4,10}]=9;
precalc [{1,5,5}]=3;
precalc [{1,5,6}]=8;
precalc [{1,5,7}]=10;
precalc [{1,5,8}]=7;
precalc [{1,5,9}]=8;
precalc [{1,5,10}]=4;
precalc [{1,6,6}]=8;
precalc [{1,6,7}]=8;
precalc [{1,6,8}]=4;
precalc [{1,6,9}]=4;
precalc [{1,6,10}]=8;
precalc [{1,7,7}]=10;
precalc [{1,7,8}]=10;
precalc [{1,7,9}]=8;
precalc [{1,7,10}]=9;
precalc [{1,8,8}]=7;
precalc [{1,8,9}]=6;
precalc [{1,8,10}]=3;
precalc [{1,9,9}]=8;
precalc [{1,9,10}]=6;
precalc [{1,10,10}]=2;
precalc [{2,2,2}]=6;
precalc [{2,2,3}]=5;
precalc [{2,2,4}]=7;
precalc [{2,2,5}]=9;
precalc [{2,2,6}]=8;
precalc [{2,2,7}]=8;
precalc [{2,2,8}]=10;
precalc [{2,2,9}]=3;
precalc [{2,2,10}]=9;
precalc [{2,3,3}]=4;
precalc [{2,3,4}]=8;
precalc [{2,3,5}]=4;
precalc [{2,3,6}]=5;
precalc [{2,3,7}]=4;
precalc [{2,3,8}]=6;
precalc [{2,3,9}]=4;
precalc [{2,3,10}]=1;
precalc [{2,4,4}]=5;
precalc [{2,4,5}]=9;
precalc [{2,4,6}]=8;
precalc [{2,4,7}]=8;
precalc [{2,4,8}]=10;
precalc [{2,4,9}]=6;
precalc [{2,4,10}]=7;
precalc [{2,5,5}]=8;
precalc [{2,5,6}]=8;
precalc [{2,5,7}]=8;
precalc [{2,5,8}]=4;
precalc [{2,5,9}]=1;
precalc [{2,5,10}]=8;
precalc [{2,6,6}]=4;
precalc [{2,6,7}]=8;
precalc [{2,6,8}]=9;
precalc [{2,6,9}]=1;
precalc [{2,6,10}]=4;
precalc [{2,7,7}]=9;

```

```

precalc [{2,7,8}]=1;
precalc [{2,7,9}]=3;
precalc [{2,7,10}]=9;
precalc [{2,8,8}]=10;
precalc [{2,8,9}]=7;
precalc [{2,8,10}]=7;
precalc [{2,9,9}]=7;
precalc [{2,9,10}]=6;
precalc [{2,10,10}]=3;
precalc [{3,3,3}]=5;
precalc [{3,3,4}]=7;
precalc [{3,3,5}]=2;
precalc [{3,3,6}]=1;
precalc [{3,3,7}]=6;
precalc [{3,3,8}]=9;
precalc [{3,3,9}]=5;
precalc [{3,3,10}]=4;
precalc [{3,4,4}]=9;
precalc [{3,4,5}]=7;
precalc [{3,4,6}]=2;
precalc [{3,4,7}]=9;
precalc [{3,4,8}]=7;
precalc [{3,4,9}]=6;
precalc [{3,4,10}]=9;
precalc [{3,5,5}]=7;
precalc [{3,5,6}]=10;
precalc [{3,5,7}]=9;
precalc [{3,5,8}]=7;
precalc [{3,5,9}]=8;
precalc [{3,5,10}]=9;
precalc [{3,6,6}]=5;
precalc [{3,6,7}]=8;
precalc [{3,6,8}]=5;
precalc [{3,6,9}]=10;
precalc [{3,6,10}]=1;
precalc [{3,7,7}]=8;
precalc [{3,7,8}]=9;
precalc [{3,7,9}]=10;
precalc [{3,7,10}]=4;
precalc [{3,8,8}]=1;
precalc [{3,8,9}]=10;
precalc [{3,8,10}]=2;
precalc [{3,9,9}]=4;
precalc [{3,9,10}]=1;
precalc [{3,10,10}]=4;
precalc [{4,4,4}]=9;
precalc [{4,4,5}]=9;
precalc [{4,4,6}]=5;
precalc [{4,4,7}]=5;
precalc [{4,4,8}]=10;
precalc [{4,4,9}]=7;
precalc [{4,4,10}]=3;
precalc [{4,5,5}]=3;
precalc [{4,5,6}]=10;
precalc [{4,5,7}]=1;
precalc [{4,5,8}]=3;
precalc [{4,5,9}]=10;
precalc [{4,5,10}]=2;
precalc [{4,6,6}]=10;
precalc [{4,6,7}]=5;
precalc [{4,6,8}]=3;
precalc [{4,6,9}]=10;
precalc [{4,6,10}]=1;
precalc [{4,7,7}]=2;
precalc [{4,7,8}]=1;
precalc [{4,7,9}]=6;
precalc [{4,7,10}]=6;
precalc [{4,8,8}]=7;
precalc [{4,8,9}]=3;
precalc [{4,8,10}]=7;
precalc [{4,9,9}]=6;
precalc [{4,9,10}]=8;
precalc [{4,10,10}]=9;
precalc [{5,5,5}]=2;

```

```

    precalc[{5,5,6}]=8;
    precalc[{5,5,7}]=9;
    precalc[{5,5,8}]=3;
    precalc[{5,5,9}]=8;
    precalc[{5,5,10}]=3;
    precalc[{5,6,6}]=2;
    precalc[{5,6,7}]=2;
    precalc[{5,6,8}]=7;
    precalc[{5,6,9}]=1;
    precalc[{5,6,10}]=2;
    precalc[{5,7,7}]=1;
    precalc[{5,7,8}]=4;
    precalc[{5,7,9}]=10;
    precalc[{5,7,10}]=8;
    precalc[{5,8,8}]=2;
    precalc[{5,8,9}]=4;
    precalc[{5,8,10}]=3;
    precalc[{5,9,9}]=3;
    precalc[{5,9,10}]=1;
    precalc[{5,10,10}]=1;
    precalc[{6,6,6}]=7;
    precalc[{6,6,7}]=4;
    precalc[{6,6,8}]=3;
    precalc[{6,6,9}]=5;
    precalc[{6,6,10}]=8;
    precalc[{6,7,7}]=2;
    precalc[{6,7,8}]=4;
    precalc[{6,7,9}]=3;
    precalc[{6,7,10}]=5;
    precalc[{6,8,8}]=7;
    precalc[{6,8,9}]=7;
    precalc[{6,8,10}]=4;
    precalc[{6,9,9}]=3;
    precalc[{6,9,10}]=8;
    precalc[{6,10,10}]=7;
    precalc[{7,7,7}]=4;
    precalc[{7,7,8}]=6;
    precalc[{7,7,9}]=8;
    precalc[{7,7,10}]=6;
    precalc[{7,8,8}]=9;
    precalc[{7,8,9}]=10;
    precalc[{7,8,10}]=6;
    precalc[{7,9,9}]=3;
    precalc[{7,9,10}]=4;
    precalc[{7,10,10}]=2;
    precalc[{8,8,8}]=1;
    precalc[{8,8,9}]=2;
    precalc[{8,8,10}]=9;
    precalc[{8,9,9}]=3;
    precalc[{8,9,10}]=1;
    precalc[{8,10,10}]=1;
    precalc[{9,9,9}]=2;
    precalc[{9,9,10}]=3;
    precalc[{9,10,10}]=6;
    precalc[{10,10,10}]=5;
}

void build_precalc123(){
    precalc3[{1,1,1}]=4;
    precalc3[{1,1,2}]=3;
    precalc3[{1,1,3}]=4;
    precalc3[{1,2,2}]=3;
    precalc3[{1,2,3}]=4;
    precalc3[{1,3,3}]=2;
    precalc3[{2,2,2}]=4;
    precalc3[{2,2,3}]=4;
    precalc3[{2,3,3}]=4;
    precalc3[{3,3,3}]=2;
}

vector <int> to_check;

void build_to_check(){

```

```

    for(int i = 1; i < 10000; ++i){
        to_check.pb(rng() % (N - 1) + 1);
    }
    sort(all(to_check));
    to_check = vector<int>(to_check.begin(), unique(to_check.begin(), to_check.end()));
}

bool in(vector<int> &a, int x){
    auto t = lower_bound(all(a), x);
    if(t == a.end())
        return 0;
    return *t == x;
}

void add(){
    int t;
    cin >> t;
    while(t--){
        int n, k;
        cin >> n >> k;
        if(n == 10){
            vector<int> a(k);
            cinm(a);
            mbort(a);
            cout << precalc[a];
        }
        else if(n == 1000000){
            cout << 337228;
        }
        else{
            vector<int> a(k);
            cinm(a);
            mbort(a);
            vector<int> tmods(1000);
            for(int i = 0; i < 1000; ++i){
                tmods[i] = rng() % 50000 + 1;
            }
            vector<vector<int>> mods(1000);
            for(auto i : a){
                for(int j = 0; j < 1000; ++j){
                    if(i % tmods[j] == 0){
                        mods[j].pb(i);
                    }
                }
            }
            bool found = 0;
            for(auto &i : mods){
                if(sz(i) == 3){
                    found = 1;
                }
            }
            assert(found);
            cout << 1;
        }
        cout << endl;
    }
}

void clear(){
    int t;
    cin >> t;
    while(t--){
        int n, k;
        cin >> n >> k;
        if(n == 10){
            vector<int> a(k + 1);
            cinm(a);
            mbort(a);
            for(int j = 0; j < 4; ++j){
                int x = a[j];
                a.erase(a.begin() + j);
                if(precalc.count(a) && precalc[a] == x){
                    coutm(a);
                    break;
                }
            }
        }
    }
}

```

```

        }
        a.insert(a.begin() + j, x);
    }
}
else if(n == 1000000){
    vector<int> a(k + 1);
    cinm(a);
    for(auto i : a){
        if(i != 337228)
            cout << i << '␣';
    }
}
else{
    vector<int> a(k + 1);
    cinm(a);
    mbort(a);
    bool found = 0;
    /*for(auto i : precalc1){
        mbort(i);
        if(in(a, i[0]) && in(a, i[1]) && in(a, i[2]) && in(a, i[3])){
            found = 1;
            a.erase(find(all(a), i[3]));
            coutm(a);
            break;
        }
    }*/
    //assert(found);
    a.erase(a.begin());
    coutm(a);
}
cout << endl;
}
}

void run() {
    build_precalc();
    string c;
    cin >> c;
    if(c == "add")
        add();
    else
        clear();
}

signed main(){
    //files("input.txt", "output.txt");
    //files("spiral.in", "spiral.out");
    iostream::sync_with_stdio(0);
    cin.tie(0);
    cout.tie(0);
    int t = 1;
    //cin >> t;
    while(t--){
        run();
        cout.flush();
    }
}

```

Task F ()

```

// #pragma GCC target ("avx2")
// #pragma GCC optimize("O3")
// #pragma GCC optimize("unroll-loops")
// #pragma GCC optimize("Ofast")
#include <bits/stdc++.h>
// #include <ext/rope>
// #include <ext/pb_ds/assoc_container.hpp>
// #include <ext/pb_ds/tree_policy.hpp>

using namespace std;
// using namespace __gnu_cxx;
// using namespace __gnu_pbds;

#define files(fin, fout) freopen(fin, "r", stdin); freopen(fout, "w", stdout)
#define foru(i, n) for(int i = 0; i < n; ++i)
#define fori(i, j, n) for(int i = j; i < n; ++i)
#define vi vector<int>
#define pii pair<int, int>
#define pb push_back
#define mp make_pair
#define mpm(a, b) make_pair(min(a, b), max(a, b))
#define mpint(a, b) ((ll)(a) * (INT_MAX) + (b))
#define all(a) (a).begin(), (a).end()
#define rall(a) (a).rbegin(), (a).rend()
#define mbort(a) sort(all(a))
#define cinm(a) for(auto &i : a) cin >> i
#define cinmm(a) for(auto &i : a) cinm(i)
#define sz(a) (int)(a).size()
#define coutm(a) for(auto &i : a) cout << i << ' '; cout << '\n'
#define coutmm(a) for(auto &i : a) coutm(i)
#define debug(x) cout << #x << ": " << x << '\n'
#define debugm(x) cout << #x << ": \n"; coutm(x)
#define ff first
#define ss second

typedef long long ll;
typedef unsigned long long ull;
typedef long double ld;

const signed inf = 1000000100;
const ll inf64 = 1ll * inf * inf;
const ll mod = 1e9 + 7;

mt19937 rng(0);

// #define int long long

void run() {
    int n;
    cin >> n;
    vector<pii> park;
    park.emplace_back(0, 0);
    park.emplace_back(1, 0);
    park.emplace_back(1, 1);
    park.emplace_back(0, 1);
    vector<pii> ans;
    ans.emplace_back(-1, 1);
    ans.emplace_back(0, 1);
    ans.emplace_back(1, 1);
    ans.emplace_back(1, 0);
    ans.emplace_back(1, -1);
    ans.emplace_back(0, -1);
    ans.emplace_back(-1, -1);
    ans.emplace_back(-1, 0);
    cout << 4 << '\n';
    for(auto i : park){
        cout << i.ff << ' ' << i.ss << '\n';
    }
    for(int i = 0; i < n; ++i){
        cout << ans[i].ff << ' ' << ans[i].ss << '\n';
    }
}

```

```
signed main(){
    // files("input.txt", "output.txt");
    // files("spiral.in", "spiral.out");
    iostream::sync_with_stdio(0);
    cin.tie(0);
    cout.tie(0);
    int t = 1;
    // cin >> t;
    while(t--){
        run();
        cout.flush();
    }
}
```