Олимпиада СПбГУ по информатике 2020/21 учебного года

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

Task A ()

```
k = int(input())
if k < 10:
    print(k)
else:
    k -= 10
    k %= 9
    if k == 0:
        print(0)
    else:
        print(k + 1)
```

Task B ()

```
n, k = map(int, input().split())
s = input()

st = set()
length = 0
ans = 1
for i in range(n):
    if length + 1 <= k and (len(st) < 3 or s[i] in st):
        length += 1
        st.add(s[i])
    else:
        ans += 1
        length = 1
        st.clear()
        st.add(s[i])

print(ans)
```

Task C ()

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

using namespace std;

#define nl '\n'
#define all(a) (a).begin(), (a).end()

typedef vector<int> vi;

void IO() {
#ifdef SEVA
    freopen("../input.txt", "r", stdin);
    freopen("../output.txt", "w", stdout);
#endif
    cin.tie(nullptr);
    cout.tie(nullptr);
    ios_base::sync_with_stdio(false);
}

const int N = 500 + 1;
const int M = 250000 + 1;

bitset<M> dp[N];
bitset<M> action[N];
int costY[N][M];

void Solve() {
    int n, x, y;
    cin >> n >> x >> y;
    vi a(n), b(n);
    for (auto& i : a)
        cin >> i;
    for (auto& i : b)
        cin >> i;
    dp[0][0] = true;
    costY[0][0] = 0;
    for (int i = 1; i <= n; i++) {
        for (int j = 0; j <= x; j++) {
            dp[i][j] = dp[i - 1][j];
            action[i][j] = false;
            costY[i][j] = costY[i - 1][j] + b[i - 1];
        }
        for (int j = a[i - 1]; j <= x; j++) {
            if (!dp[i - 1][j - a[i - 1]])
                continue;
            if (!dp[i][j] || (dp[i][j] && costY[i - 1][j - a[i - 1]] < costY[i][j])) {
                dp[i][j] = true;
                action[i][j] = true;
                costY[i][j] = costY[i - 1][j - a[i - 1]];
            }
        }
    }
    for (int j = 0; j <= x; j++) {
        if (!dp[n][j])
            continue;
        if (costY[n][j] > y)
            continue;
        string ans;
        int sumX = j;
        int sumY = costY[n][j];
        for (int i = n; i > 0; i--) {
            if (action[i][sumX]) {
                sumX -= a[i - 1];
                ans += 'x';
            }
            else {
                sumY -= b[i - 1];
                ans += 'y';
            }
        }
        assert(sumX == 0 && sumY == 0);
        reverse(all(ans));
```

```
            cout << ans << nl;
            return;
        }
        cout << -1 << nl;
}

signed main() {
    IO();
    Solve();
    return 0;
}
```

Task D ()

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

using namespace std;

#define nl '\n'
#define all(a) (a).begin(), (a).end()
#define minq(a, b) (a) = min(a, b)
#define maxq(a, b) (a) = max(a, b)

typedef vector<int> vi;

void IO() {
#ifdef SEVA
    freopen("../input.txt", "r", stdin);
    freopen("../output.txt", "w", stdout);
#endif
    cin.tie(nullptr);
    cout.tie(nullptr);
    ios_base::sync_with_stdio(false);
}

const int N = int(2e5) + 10;

bool bs[N];

void Solve() {
    int n;
    cin >> n;
    string s;
    cin >> s;
    for (int i = 0; i < 2 * n; i++) {
        bs[i] = s[i] == '(' || s[i] == ')';
    }
    stack<bool> st;
    for (int i = 0; i < 2 * n; i++) {
        if (st.empty() || st.top() != bs[i])
            st.emplace(bs[i]);
        else
            st.pop();
    }
    cout << st.size() / 2 << nl;
}

signed main() {
    IO();
    Solve();
    return 0;
}
```

Task E ()

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

using namespace std;

typedef unsigned long long ull;

#define int ull

#define nl '\n'
#define all(a) (a).begin(), (a).end()
#define minq(a, b) (a) = min(a, b)
#define maxq(a, b) (a) = max(a, b)
#define sz(a) int((a).size())

typedef long long ll;
typedef vector<int> vi;

mt19937 rnd(2330);

void IO() {
#ifdef SEVA
    freopen("../input.txt", "r", stdin);
    freopen("../output.txt", "w", stdout);
#else
    cin.tie(nullptr);
    cout.tie(nullptr);
    ios_base::sync_with_stdio(false);
#endif
}

const int NUM = 125449;

void Add1(int n, int k, set<int>& st) {
    assert(n == int(1e6));
    cout << NUM << nl;
}

void Clear1(int n, int k, set<int>& st) {
    assert(n == int(1e6));
    for (auto& i : st) {
        if (i != NUM)
            cout << i << '_';
    }
    cout << nl;
}

const int N = 10;

int bro[1 << N];

void Add2(int n, int k, set<int>& st) {
    assert(n == 10);
    assert(k == 3);
    int m1 = 0;
    for (auto& i : st)
        m1 |= 1 << (i - 1);
    assert(__builtin_popcount(m1) == 3);
    int m2 = bro[m1];
    assert(__builtin_popcount(m2) == 4);
    int m = m1 ^ m2;
    assert(__builtin_popcount(m) == 1);
    for (int i = 0; i < N; i++) {
        if (m >> i & 1) {
            cout << i + 1 << nl;
        }
    }
}

void Clear2(int n, int k, set<int>& st) {
    assert(n == 10);
    assert(k == 3);
    int m1 = 0;
```

```
    for (auto& i : st)
        m1 |= 1 << (i - 1);
    assert(__builtin_popcount(m1) == 4);
    int m2 = bro[m1];
    assert(__builtin_popcount(m2) == 3);
    for (int i = 0; i < N; i++) {
        if (m2 >> i & 1) {
            cout << i + 1 << '␣';
        }
    }
    cout << nl;
}

void MakeBro() {
    for (int m1 = 0; m1 < (1 << N); m1++) {
        if (__builtin_popcount(m1) != 4)
            continue;
        vi cands;
        for (int i = 0; i < N; i++) {
            if (!(m1 >> i & 1))
                continue;
            int m2 = m1 ^ (1 << i);
            assert(__builtin_popcount(m2) == 3);
            if (bro[m2] != 0)
                continue;
            cands.emplace_back(m2);
        }
        if (cands.empty())
            continue;
        int m2 = cands[rnd() % sz(cands)];
        bro[m2] = m1;
        bro[m1] = m2;
    }
}

const int P = 2000003;

int Pow(int a, int n) {
    int ans = 1;
    while (n) {
        if (n & 1) {
            ans *= a;
        }
        a *= a;
        n >>= 1;
    }
    return ans;
}

void HashAdd(int& h, int el) {
    h += Pow(P, el);
}

void HashDel(int& h, int el) {
    h -= Pow(P, el);
}

int Hash(set<int>& st) {
    int h = 0;
    for (auto& i : st) {
        HashAdd(h, i);
    }
    return h;
}

int MakeNum(const set<int>& st, int h, int n) {
    mt19937_64 r(h + 1);
    int cnt = 0;
    int ans;
    for (;;) {
        ans = r() % n + 1;
        if (st.count(ans)) {
            cnt = 0;
        }
```

```
        cnt++;
        if (cnt == 5)
            break;
    }
    return ans;
}

void Add3(int n, int m, set<int>& st) {
    int h = Hash(st);
    int num = MakeNum(st, h, n);
    cout << num << nl;
}

void Clear3(int n, int m, set<int>& st) {
    auto st2 = st;
    int h = Hash(st);
    for (auto& i : st2) {
        HashDel(h, i);
        st.erase(i);
        int num = MakeNum(st, h, n);
        if (num == i) {
            for (auto& i : st) {
                cout << i << ' ';
            }
            cout << nl;
            return;
        }
        st.emplace(i);
        HashAdd(h, i);
    }
    assert(false);
}

void Add() {
    int n, k;
    cin >> n >> k;
    set<int> st;

    for (int i = 0; i < k; i++) {
        int a;
        cin >> a;
        st.emplace(a);
    }
    if (n == (int)1e6) {
        Add1(n, k, st);
    }
    else if (n == 10) {
        Add2(n, k, st);
    }
    else if (n == (int)1e5) {
        Add3(n, k, st);
    }
}

void Clear() {
    int n, k;
    cin >> n >> k;
    set<int> st;
    for (int j = 0; j <= k; j++) {
        int a;
        cin >> a;
        st.emplace(a);
    }
    if (n == int(1e6)) {
        Clear1(n, k, st);
    }
    else if (n == 10) {
        Clear2(n, k, st);
    }
    else if (n == (int)1e5) {
        Clear3(n, k, st);
    }
}
```

```
void Solve() {
    string type;
    cin >> type;
    int t;
    cin >> t;
    MakeBro();
    for (int i = 0; i  < t; i++) {
        if (type == "add")
            Add();
        else
            Clear();
    }
}

signed main() {
    IO();
    Solve();
    return 0;
}
```

Task F ()

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

using namespace std;

#define nl '\n'
#define all(a) (a).begin(), (a).end()
#define minq(a, b) (a) = min(a, b)
#define maxq(a, b) (a) = max(a, b)

typedef vector<int> vi;

void IO() {
#ifdef SEVA
    freopen("../input.txt", "r", stdin);
    freopen("../output.txt", "w", stdout);
#endif
    cin.tie(nullptr);
    cout.tie(nullptr);
    ios_base::sync_with_stdio(false);
}

mt19937 rnd(23432);

void Solve() {
    int n;
    cin >> n;
    cout << 4 << nl;
    cout << 0 << ' ' << 0 << nl;
    cout << 1 << ' ' << 0 << nl;
    cout << 1 << ' ' << 1 << nl;
    cout << 0 << ' ' << 1 << nl;
    for (int i = -1; i <= 1 && n > 0; i++) {
        for (int j = -1; j <= 1 && n > 0; j++) {
            if (i == 0 && j == 0)
                continue;
            n--;
            cout << i << ' ' << j << nl;
        }
    }
}

signed main() {
    IO();
    Solve();
    return 0;
}
```