Олимпиада СПбГУ по информатике 2020/21 учебного года

A	B	C	D	E	F	Sum
100	100	100	100	20	25	445

Task A ()

```
#define _CRT_SECURE_NO_WARNINGS

/*#pragma GCC optimize("Ofast")
#pragma GCC optimize("unroll-loops")*/

#include <iostream>
#include <iomanip>
#include <cmath>
#include <algorithm>
#include <string>
#include <vector>
#include <stack>
#include <deque>
#include <queue>
#include <map>
#include <set>
#include <cstring>
#include <cstdio>
#include <climits>
#include <unordered_map>
#include <unordered_set>

using namespace std;

typedef long long ll;
typedef long double ld;
typedef unsigned long long ull;
typedef pair <ll, ll> pll;
typedef vector <ll> vec;
typedef vector <pll> vecp;
typedef vector <ld> vecd;
typedef vector <vector <ll>> matrix;

const ld PI = 3.141592653589793;
const ld eps = 1e-9;
const ll INF = 100000000002;
const ll MOD = 1e9 + 7;

#define pb push_back
#define mp make_pair
#define all(v) v.begin(), v.end()
#define fast_io ios::sync_with_stdio(false), cin.tie(0), cout.tie(0)
#define fixed_out(x) fixed << setprecision(x)

int main() {
    fast_io;
    ull k;
    cin >> k;
    if (k == 1) {
        cout << 1;
    }
    else if (k % 9 == 1) {
        cout << 0;
    }
    else if (k % 9 == 0) {
        cout << 9;
```

```
    }
    else {
        cout << k % 9;
    }
    return 0;
}
```

Task B ()

```cpp
#define _CRT_SECURE_NO_WARNINGS

/*#pragma GCC optimize("Ofast")
#pragma GCC optimize("unroll-loops")*/

#include <iostream>
#include <iomanip>
#include <cmath>
#include <algorithm>
#include <string>
#include <vector>
#include <stack>
#include <deque>
#include <queue>
#include <map>
#include <set>
#include <cstring>
#include <cstdio>
#include <climits>
#include <unordered_map>
#include <unordered_set>

using namespace std;

typedef long long ll;
typedef long double ld;
typedef unsigned long long ull;
typedef pair <ll, ll> pll;
typedef vector <ll> vec;
typedef vector <pll> vecp;
typedef vector <ld> vecd;
typedef vector <vector <ll>> matrix;

const ld PI = 3.141592653589793;
const ld eps = 1e-9;
const ll INF = 100000000002;
const ll MOD = 1e9 + 7;

#define pb push_back
#define mp make_pair
#define all(v) v.begin(), v.end()
#define fast_io ios::sync_with_stdio(false), cin.tie(0), cout.tie(0)
#define fixed_out(x) fixed << setprecision(x)

int main() {
    fast_io;
    ll n, k, cnt = 0, ans = 1;
    string s;
    cin >> n >> k >> s;
    map <char, ll> m;
    for (int i = 0; i < n; ++i) {
        ++cnt;
        ++m[s[i]];
        if ((cnt > k) || (m.size() > 3)) {
            ++ans;
            cnt = 1;
            m.clear();
            ++m[s[i]];
        }
    }
    cout << ans;
    return 0;
}
```

Task C ()

```
#define _CRT_SECURE_NO_WARNINGS

/*#pragma GCC optimize("Ofast")
#pragma GCC optimize("unroll-loops")*/

#include <iostream>
#include <iomanip>
#include <cmath>
#include <algorithm>
#include <string>
#include <vector>
#include <stack>
#include <deque>
#include <queue>
#include <map>
#include <set>
#include <cstring>
#include <cstdio>
#include <climits>
#include <unordered_map>
#include <unordered_set>

using namespace std;

typedef int ll;
typedef long double ld;
typedef unsigned long long ull;
typedef pair <ll, ll> pll;
typedef vector <ll> vec;
typedef vector <pll> vecp;
typedef vector <ld> vecd;
typedef vector <vector <ll>> matrix;

const ld PI = 3.141592653589793;
const ld eps = 1e-9;
const ll INF = 100000000002;
const ll MOD = 1e9 + 7;

#define pb push_back
#define mp make_pair
#define all(v) v.begin(), v.end()
#define fast_io ios::sync_with_stdio(false), cin.tie(0), cout.tie(0)
#define fixed_out(x) fixed << setprecision(x)

struct item {
        ll v, w;
};

int main() {
    fast_io;
        ll n, x, y;
        vec used;
        vector <item> v;
        matrix v_dp, w_dp;
        cin >> n >> x >> y;
        v.resize(n);
        used.resize(n);
        v_dp.resize(n + 1);
        for (int i = 0; i < n; ++i) {
                v_dp[i].resize(x + 1);
                cin >> v[i].v;
        }
        v_dp[n].resize(x + 1);
        for (int i = 0; i < n; ++i) {
                cin >> v[i].w;
        }
        for (int i = 0; i < x; ++i) {
                v_dp[0][i] = 0;
        }
        for (int i = 1; i <= n; ++i) {
                for (int j = 0; j <= x; ++j) {
                        if (j - v[i - 1].v >= 0) {
```

```
                                v_dp[i][j] = max(v_dp[i - 1][j], v_dp[i - 1][j - v[i - 1].v] + v[i
                                    - 1].w);
                        }
                        else {
                                v_dp[i][j] = v_dp[i - 1][j];
                        }
                }
        }
        ll i = n, j = x;
        while (v_dp[i][j] != 0) {
                if (v_dp[i][j] == v_dp[i - 1][j]) {
                        --i;
                }
                else {
                        --i;
                        j -= v[i].v;
                        used[i] = 1;
                }
        }
        ll w_sum = 0;
        for (int i = 0; i < n; ++i) {
                if (!used[i]) {
                        w_sum += v[i].w;
                }
        }
        if (w_sum <= y) {
                for (int i = 0; i < n; ++i) {
                        if (used[i]) {
                                cout << 'x';
                        }
                        else {
                                cout << 'y';
                        }
                }
        }
        else {
                cout << -1;
        }
    return 0;
}
```

Task D ()

```
#define _CRT_SECURE_NO_WARNINGS

/*#pragma GCC optimize("Ofast")
#pragma GCC optimize("unroll-loops")*/

#include <iostream>
#include <iomanip>
#include <cmath>
#include <algorithm>
#include <string>
#include <vector>
#include <stack>
#include <deque>
#include <queue>
#include <map>
#include <set>
#include <cstring>
#include <cstdio>
#include <climits>
#include <unordered_map>
#include <unordered_set>

using namespace std;

typedef int ll;
typedef long double ld;
typedef unsigned long long ull;
typedef pair <ll, ll> pll;
typedef vector <ll> vec;
typedef vector <pll> vecp;
typedef vector <ld> vecd;
typedef vector <vector <ll>> matrix;

const ld PI = 3.141592653589793;
const ld eps = 1e-9;
const ll INF = 100000000002;
const ll MOD = 1e9 + 7;

#define pb push_back
#define mp make_pair
#define all(v) v.begin(), v.end()
#define fast_io ios::sync_with_stdio(false), cin.tie(0), cout.tie(0)
#define fixed_out(x) fixed << setprecision(x)

bool check(char a, char b) {
        return ((((a == '(') || (a == ')')) && ((b == '(') || (b == ')'))) || (((a == '[') || (a
            == ']')) && ((b == '[') || (b == ']'))));
}

int main() {
    fast_io;
        ll n, cnt = 0;
        string s;
        cin >> n >> s;
        bool flag = 1;
        while (flag) {
                flag = 0;
                for (int i = 1; i < s.size(); ++i) {
                        if (check(s[i], s[i - 1])) {
                                flag = 1;
                                s.erase(s.begin() + i - 1, s.begin() + i + 1);
                                break;
                        }
                }
        }
        cout << s.size() / 2;
    return 0;
}
```

Task E ()

```
#define _CRT_SECURE_NO_WARNINGS

/*#pragma GCC optimize("Ofast")
#pragma GCC optimize("unroll-loops")*/

#include <iostream>
#include <iomanip>
#include <cmath>
#include <algorithm>
#include <string>
#include <vector>
#include <stack>
#include <deque>
#include <queue>
#include <map>
#include <set>
#include <cstring>
#include <cstdio>
#include <climits>
#include <unordered_map>
#include <unordered_set>

using namespace std;

typedef int ll;
typedef long double ld;
typedef unsigned long long ull;
typedef pair <ll, ll> pll;
typedef vector <ll> vec;
typedef vector <pll> vecp;
typedef vector <ld> vecd;
typedef vector <vector <ll>> matrix;

const ld PI = 3.141592653589793;
const ld eps = 1e-9;
const ll INF = 100000000002;
const ll MOD = 1e9 + 7;

#define pb push_back
#define mp make_pair
#define all(v) v.begin(), v.end()
#define fast_io ios::sync_with_stdio(false), cin.tie(0), cout.tie(0)
#define fixed_out(x) fixed << setprecision(x)

int main() {
    ll num = 62735;
    fast_io;
    string s;
    cin >> s;
    if (s == "add") {
        ll t;
        cin >> t;
        while (t--) {
            ll n, k;
            vec v;
            cin >> n >> k;
            v.resize(k);
            for (int i = 0; i < k; ++i) {
                cin >> v[i];
            }
            if (n == 10) {
                cout << 4;
            }
            else {
                cout << num << "\n";
            }
        }
    }
    else {
        ll t;
        cin >> t;
        while (t--) {
```

```
            ll n, k;
            vec v;
            cin >> n >> k;
            v.resize(k + 1);
            for (int i = 0; i < k + 1; ++i) {
                cin >> v[i];
            }
            if (n == 10) {
                cout << "2 3 7";
            }
            else {
                for (int i = 0; i < k + 1; ++i) {
                    if (v[i] != num) {
                        cout << v[i] << ' ';
                    }
                }
                cout << "\n";
            }
        }
    }
    return 0;
}
```

Task F ()

```
#define _CRT_SECURE_NO_WARNINGS

/*#pragma GCC optimize("Ofast")
#pragma GCC optimize("unroll-loops")*/

#include <iostream>
#include <iomanip>
#include <cmath>
#include <algorithm>
#include <string>
#include <vector>
#include <stack>
#include <deque>
#include <queue>
#include <map>
#include <set>
#include <cstring>
#include <cstdio>
#include <climits>
#include <unordered_map>
#include <unordered_set>

using namespace std;

typedef int ll;
typedef long double ld;
typedef unsigned long long ull;
typedef pair <ll, ll> pll;
typedef vector <ll> vec;
typedef vector <pll> vecp;
typedef vector <ld> vecd;
typedef vector <vector <ll>> matrix;

const ld PI = 3.141592653589793;
const ld eps = 1e-9;
const ll INF = 100000000002;
const ll MOD = 1e9 + 7;

#define pb push_back
#define mp make_pair
#define all(v) v.begin(), v.end()
#define fast_io ios::sync_with_stdio(false), cin.tie(0), cout.tie(0)
#define fixed_out(x) fixed << setprecision(x)

int main() {
    ll n;
    cin >> n;
    cout << "12 4 1 4 0 5 0 5 1 6 1 6 2 5 2 5 3 4 3 4 2 3 2 3 1\n";
    string ans[8] = { "3 0", "0 3", "-3 0", "0 -3", "-2 -2", "-2 2", "2 -2", "2 2" };
    for (int i = 0; i < n; ++i) {
        cout << ans[i] << "\n";
    }
    return 0;
}
```