Олимпиада СПбГУ по информатике 2020/21 учебного года

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

Task A ()

```cpp
// #define _GLIBCXX_DEBUG
#include <bits/stdc++.h>
using namespace std;

signed main() {
        #ifdef LOCAL
                freopen("input.txt", "r", stdin);
                freopen("output.txt", "w", stdout);
                cout << fixed << setprecision(3);
        #else
                ios_base::sync_with_stdio(false);
                cin.tie(nullptr);
                cout.tie(nullptr);
                cout << fixed << setprecision(15);
        #endif
        string s = "234567890";
        int k;
        cin >> k;
        if (k <= 10) {
                cout << k % 10;
        } else {
                k -= 11;
                cout << s[k % s.size()];
        }
        #ifdef LOCAL
                cerr << "proc␣time:␣" << 1.0 * clock() / CLOCKS_PER_SEC << "s␣␣␣␣";
        #endif
        return 0;
}
```

Task B ()

```
// #define _GLIBCXX_DEBUG
#include <bits/stdc++.h>
using namespace std;

signed main() {
        #ifdef LOCAL
                freopen("input.txt", "r", stdin);
                freopen("output.txt", "w", stdout);
                cout << fixed << setprecision(3);
        #else
                ios_base::sync_with_stdio(false);
                cin.tie(nullptr);
                cout.tie(nullptr);
                cout << fixed << setprecision(15);
        #endif
        int n, k;
        cin >> n >> k;
        string s;
        cin >> s;
        vector<int> cnt(26);
        int diff = 0, len = 0, ans = 1, C = 1;
        for (int i = 0; i < n; ++i) {
                int c = (s[i] - 'a');
                if ((cnt[c] != C && diff == 3) || len == k) {
                        ++C;
                        ++ans;
                        len = 0;
                        diff = 0;
                }
                if (cnt[c] != C) {
                        cnt[c] = C;
                        ++diff;
                }
                ++len;
        }
        cout << ans;
        #ifdef LOCAL
                cerr << "proc␣time:␣" << 1.0 * clock() / CLOCKS_PER_SEC << "s␣␣␣␣";
        #endif
        return 0;
}
```

Task C ()

```
// #define _GLIBCXX_DEBUG
#include <bits/stdc++.h>
using namespace std;

const int MAXN = 501;
const int MAXX = 250001;
int a[MAXN], b[MAXN];
int dp[MAXN][MAXX];

signed main() {
        #ifdef LOCAL
                freopen("input.txt", "r", stdin);
                freopen("output.txt", "w", stdout);
                cout << fixed << setprecision(3);
        #else
                ios_base::sync_with_stdio(false);
                cin.tie(nullptr);
                cout.tie(nullptr);
                cout << fixed << setprecision(15);
        #endif
        fill(&dp[0][0], &dp[0][0] + MAXN * MAXX, 1e9);
        int n, X, Y;
        cin >> n >> X >> Y;
        for (int i = 0; i < n; ++i) cin >> a[i];
        for (int i = 0; i < n; ++i) cin >> b[i];
        vector<vector<bool>> p(n + 1, vector<bool>(X + 1, false));
        dp[0][0] = 0;
        for (int i = 0; i < n; ++i) {
                for (int j = X; j >= 0; --j) {
                        if (dp[i + 1][j] > dp[i][j] + b[i]) {
                                dp[i + 1][j] = dp[i][j] + b[i];
                                p[i + 1][j] = 1;
                        }
                        int newa = j + a[i];
                        if (newa <= X) {
                                if (dp[i + 1][newa] > dp[i][j]) {
                                        dp[i + 1][newa] = dp[i][j];
                                        p[i + 1][newa] = 0;
                                }
                        }
                }
        }
        for (int j = X; j >= 0; --j) {
                if (dp[n][j] <= Y) {
                        string ans(n, 'z');
                        for (int i = n; i >= 1; --i) {
                                if (p[i][j] == 1) {
                                        ans[i - 1] = 'y';
                                } else {
                                        ans[i - 1] = 'x';
                                        j -= a[i - 1];
                                }
                        }
                        cout << ans;
                        return 0;
                }
        }
        cout << -1;
        #ifdef LOCAL
                cerr << "proc␣time:␣" << 1.0 * clock() / CLOCKS_PER_SEC << "s␣␣␣␣";
        #endif
        return 0;
}
```

Task D ()

```
// #define _GLIBCXX_DEBUG
#include <bits/stdc++.h>
using namespace std;

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

char type(char a) {
        if (a == '[' || a == ']') return '[';
        else return '(';
}

signed main() {
        #ifdef LOCAL
                freopen("input.txt", "r", stdin);
                freopen("output.txt", "w", stdout);
                cout << fixed << setprecision(3);
        #else
                ios_base::sync_with_stdio(false);
                cin.tie(nullptr);
                cout.tie(nullptr);
                cout << fixed << setprecision(15);
        #endif
        int n;
        string s;
        cin >> n >> s;
        vector<char> q;
        for (int i = 0; i < 2 * n; ++i) {
                if (!q.empty() && type(q.back()) == type(s[i])) {
                        q.pop_back();
                } else {
                        q.emplace_back(s[i]);
                }
        }
        int ans = 0;
        while (!q.empty()) {
                char b = q.back();
                q.pop_back();
                char a = q.back();
                q.pop_back();
                if (type(a) != type(b)) ++ans;
        }
        cout << ans;
        #ifdef LOCAL
                cerr << "proc␣time:␣" << 1.0 * clock() / CLOCKS_PER_SEC << "s␣␣␣␣";
        #endif
        return 0;
}
```

Task E ()

```
// #define _GLIBCXX_DEBUG
#include <bits/stdc++.h>
using namespace std;

long long secret = 69133722848;

void add1(int n, int k) {
        vector<int> a(k);
        for (int i = 0; i < k; ++i) {
                cin >> a[i];
        }
        cout << secret % n + 1 << '\n';
}

void clear1(int n, int k) {
        vector<int> a(k + 1);
        for (int i = 0; i < k + 1; ++i) {
                cin >> a[i];
                if (a[i] != secret % n + 1) cout << a[i] << '␣';
        }
        cout << '\n';
}


void add2(int n, int k) {
        vector<int> a(k);
        for (int i = 0; i < k; ++i) {
                cin >> a[i];
        }
        cout << 4 << '\n';
}

void clear2(int n, int k) {
        vector<int> a(k + 1);
        for (int i = 0; i < k + 1; ++i) {
                cin >> a[i];
                if (a[i] != 4) cout << a[i] << '␣';
        }
        cout << '\n';
}


void add3(int n, int k) {
        vector<int> a(k);
        for (int i = 0; i < k; ++i) {
                cin >> a[i];
        }
        cout << secret % n + 1 << '\n';
}

void clear3(int n, int k) {
        vector<int> a(k + 1);
        for (int i = 0; i < k + 1; ++i) {
                cin >> a[i];
                if (a[i] != secret % n + 1) cout << a[i] << '␣';
        }
        cout << '\n';
}


signed main() {
        #ifdef LOCAL
                freopen("input.txt", "r", stdin);
                freopen("output.txt", "w", stdout);
                cout << fixed << setprecision(3);
        #else
                ios_base::sync_with_stdio(false);
                cin.tie(nullptr);
                cout.tie(nullptr);
                cout << fixed << setprecision(15);
        #endif
        string type;
        cin >> type;
        int T;
        cin >> T;
        while (T --> 0) {
                int n, k;
                cin >> n >> k;
                if (type == "add") {
                        if (n == 1000000) add1(n, k);
                        else if (n == 10) add2(n, k);
                        else add3(n, k);
                } else {
                        if (n == 1000000) clear1(n, k);
                        else if (n == 10) clear2(n, k);
                        else clear3(n, k);
                }
        }
        #ifdef LOCAL
                cerr << "proc␣time:␣" << 1.0 * clock() / CLOCKS_PER_SEC << "s␣␣␣␣";
        #endif
        return 0;
}
```

Task F ()

```
// #define _GLIBCXX_DEBUG
#include <bits/stdc++.h>
using namespace std;


signed main() {
        #ifdef LOCAL
                freopen("input.txt", "r", stdin);
                freopen("output.txt", "w", stdout);
                cout << fixed << setprecision(3);
        #else
                ios_base::sync_with_stdio(false);
                cin.tie(nullptr);
                cout.tie(nullptr);
                cout << fixed << setprecision(15);
        #endif
        int n;
        cin >> n;

        cout << 4 << '\n';
        cout << "0␣0\n1␣0\n1␣1\n0␣1\n";
        vector<pair<int, int>> ans = {
                {0, 1},
                {1, 1},
                {1, 0},
                {1, -1},
                {0, -1},
                {-1, -1},
                {-1, 0},
                {-1, 1}
        };
        for (int i = 0; i < n; ++i) cout << ans[i].first << '␣' << ans[i].second << '\n';
        #ifdef LOCAL
                cerr << "proc␣time:␣" << 1.0 * clock() / CLOCKS_PER_SEC << "s␣␣␣␣";
        #endif
        return 0;
}
```