Олимпиада СПбГУ по информатике 2020/21 учебного года

A	B	C	D	E	F	Sum
100	100	100	100	20	15	435

Task A ()

```cpp
#include <iostream>


using namespace std;

int main() {
    int n;
    cin >> n;

    /*long long to_add = 0;
    long long curr = 0;

    for (int i = 1; i <= n; ++i) {
        curr = i % 10;
        curr += to_add;
        if (curr >= 10) {
            to_add = curr / 10;
        }
        curr %= 10;
    }

    cout << curr;*/

    /*if (n < 10) {
        cout << n;
        return 0;
    }

    cout << (n - 1) % 9;*/

    int a[10];
    a[0] = 0;
    a[1] = 2;
    a[2] = 3;
    a[3] = 4;
    a[4] = 5;
    a[5] = 6;
    a[6] = 7;
    a[7] = 8;
    a[8] = 9;

    if (n < 10) {
        cout << n;
        return 0;
    }

    cout << a[(n - 1) % 9];

    return 0;
}
```

Task B ()

```cpp
#include <iostream>
#include <string>
#include <set>

using namespace std;

int main() {
    int n, k;
    cin >> n >> k;

    string s;
    cin >> s;

    set<char> letters;
    int length = 1;
    int answer = 1;

    for (int i = 0; i < n; ++i, ++length) {
        if (length > k) {
            ++answer;
            length = 1;
            letters.clear();
        }

        letters.insert(s[i]);

        if (letters.size() > 3) {
            ++answer;
            letters.clear();
            letters.insert(s[i]);
            length = 1;
        }

    }
    cout << answer;

    return 0;
}
```

Task C ()

```cpp
#include <iostream>
#include <vector>

using namespace std;

int main() {
    int n, x, y;
    cin >> n >> x >> y;

    vector<int> V(n);
    for (auto &i : V)
        cin >> i;

    vector<int> M(n);
    for (auto &i : M)
        cin >> i;


    vector<vector<int> > dp(n + 1, vector<int>(x + 1, 0));

    for (int i = 1; i <= n; ++i) {
        for (int j = 1; j <= x; ++j) {
            dp[i][j] = max(dp[i - 1][j], dp[i][j - 1]);

            if (j - V[i - 1] >= 0)
                dp[i][j] = max(dp[i][j], dp[i - 1][j - V[i - 1]] + M[i - 1]);
        }
    }

    vector<int> used;

    int a, b;
    a = n;
    b = x;

    while (a > 0 && b > 0) {
        if (b - V[a - 1] >= 0) {
            if (dp[a - 1][b - V[a - 1]] + M[a - 1] >= dp[a - 1][b]) {
                used.push_back(a);
                --a;
                b -= V[a];
            } else {
                --a;
            }
            continue;
        }
        --a;

    }
    int test = 0;
    for (int i = 0; i < n; ++i) {
        test += M[i];
    }
    for (auto i : used) {
        test -= M[i - 1];
    }

    if (test <= y) {
        vector<char> answer(n, 'y');
        for (auto i : used)
            answer[i - 1] = 'x';

        for (auto i : answer)
            cout << i;
    } else {
        cout << -1;
    }
}
```

Task D ()

```cpp
#include <iostream>
#include <stack>

using namespace std;

int main() {
    int n;
    cin >> n;

    string s;
    cin >> s;

    stack<char> st;

    for (int i = 0; i < 2 * n; ++i) {
        char type = 1;
        if (s[i] == '[' || s[i] == ']')
            type = 2;
        if (st.empty()) {
            st.push(type);
            continue;
        }

        if (st.top() == type)
            st.pop();
        else
            st.push(type);
    }
    cout << st.size() / 2;
}
```

Task E ()

```cpp
#include <iostream>
#include <vector>
#include <string>

using namespace std;

const int abcd = 10000;

int main() {
    int test_numb = abcd;
    string a;
    cin >> a;
    if (a == "add") {
        int t;
        cin >> t;
        while (t--) {
            int n, k;
            cin >> n >> k;

            if (test_numb > n) {
                test_numb = n;
            }

            vector<int> mass(k);
            for (auto &i:mass)
                cin >> i;
            cout << test_numb << endl;
            test_numb = abcd;
        }
    } else {
        int t;
        cin >> t;
        while (t--) {
            int n, k;
            cin >> n >> k;

            if (test_numb > n) {
                test_numb = n;
            }
            vector<int> mass;
            for (int i = 0; i <= k; ++i) {
                int number;
                cin >> number;
                if (number != test_numb)
                    mass.push_back(number);
            }
            for (auto i:mass)
                cout << i << "␣";
            cout << endl;
            test_numb = abcd;
        }
    }
}
```

Task F ()

```
#include <iostream>
#include <vector>

using namespace std;

struct point2d {
    int x, y;

    point2d(int x, int y) {
        this->x = x;
        this->y = y;
    }
};

int main() {
    int n;
    cin >> n;

    vector<point2d> start_figure;
    start_figure.push_back(point2d(0, 0));
    start_figure.push_back(point2d(4, 0));
    start_figure.push_back(point2d(2, 4));

    /*vector<vector<point2d> > figures(7);

    figures[0].push_back(start_figure[0]);
    figures[0].push_back(point2d(start_figure[0].x - 2, start_figure[0].y - 4));
    figures[0].push_back(point2d(start_figure[0].x + 2, start_figure[0].y - 4));

    figures[1].push_back(start_figure[1]);
    figures[1].push_back(point2d(start_figure[1].x - 2, start_figure[1].y - 4));
    figures[1].push_back(point2d(start_figure[1].x + 2, start_figure[1].y - 4));

    figures[3].push_back(start_figure[0]);
    figures[3].push_back(point2d(start_figure[0].x - 4, start_figure[0].y));
    figures[3].push_back(point2d(start_figure[0].x - 2, start_figure[0].y + 4));

    figures[4].push_back(start_figure[1]);
    figures[4].push_back(point2d(start_figure[1].x + 4, start_figure[1].y));
    figures[4].push_back(point2d(start_figure[1].x + 2, start_figure[1].y + 4));

    figures[5].push_back(start_figure[2]);
    figures[5].push_back(point2d(start_figure[2].x + 4, start_figure[2].y));
    figures[5].push_back(point2d(start_figure[2].x + 2, start_figure[2].y + 4));

    figures[6].push_back(start_figure[2]);
    figures[6].push_back(point2d(start_figure[2].x - 4, start_figure[2].y));
    figures[6].push_back(point2d(start_figure[2].x - 2, start_figure[2].y + 4));

    figures.erase(figures.begin() + 2);
    cout << 3 << endl;
    for (int i = 0; i < n; ++i) {
        for (auto j : figures[i]) {
            cout << j.x << " " << j.y << endl;
        }
    }*/

    cout << 3 << endl;
    for (auto &i:start_figure)
        cout << i.x << "␣" << i.y << endl;

    vector<pair<int, int> > answer(6);
    answer[0] = {-2, -4};
    answer[1] = {2, -4};
    answer[2] = {-4, 0};
    answer[3] = {4, 0};
    answer[4] = {2, 4};
    answer[5] = {-2, 4};

    for (int i = 0; i < n; ++i) {
        cout << answer[i].first << "␣" << answer[i].second << endl;
    }
```

```
}
```