Олимпиада СПбГУ по информатике 2020/21 учебного года

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

Task A ()

```
#define _CRT_SECURE_NO_WARNINGS

#include <set>
#include <unordered_set>
#include <map>
#include <iostream>
#include <vector>
#include <algorithm>
#include <queue>
#include <tuple>
#include <random>
#include <ctime>
#include <string>

using namespace std;

int main()
{
#ifdef _DEBUG
        freopen("input.txt", "r", stdin);
#endif
        ios::sync_with_stdio(0);
        cin.tie(0);
        cout.tie(0);
        srand(time(0));

        vector<int> seq = { 0, 2, 3, 4, 5, 6, 7, 8, 9 };
        int k;
        cin >> k;
        if (k < 10)
                cout << k;
        else
        {
                k -= 10;
                cout << seq[k % 9];
        }
        /*int prev = 0;
        for (int i = 1; i <= k; i++)
        {
                cout << i << ' ' << (i + prev) % 10 << '\n';
                prev = (i + prev) / 10;
        }*/
        return 0;
}
```

Task B ()

```
#define _CRT_SECURE_NO_WARNINGS

#include <set>
#include <unordered_set>
#include <map>
#include <iostream>
#include <vector>
#include <algorithm>
#include <queue>
#include <tuple>
#include <random>
#include <ctime>
#include <string>

using namespace std;

int main()
{
#ifdef _DEBUG
        freopen("input.txt", "r", stdin);
#endif
        ios::sync_with_stdio(0);
        cin.tie(0);
        cout.tie(0);
        srand(time(0));

        int n, k;
        cin >> n >> k;
        string s;
        cin >> s;
        vector<bool> mark(26, 0);
        int q = 0;
        int len = 0;
        int ans = 1;
        for (char c : s)
        {
                if (len + 1 > k || (mark[c - 'a'] == 0 && q == 3))
                {
                        ans++;
                        mark.clear();
                        mark.resize(26, 0);
                        len = 0;
                        q = 0;
                }
                if (mark[c - 'a'] == 0)
                        q++;
                len++, mark[c - 'a'] = 1;
        }
        cout << ans;
        return 0;
}
```

Task C ()

```
#define _CRT_SECURE_NO_WARNINGS

#include <set>
#include <unordered_set>
#include <map>
#include <iostream>
#include <vector>
#include <algorithm>
#include <queue>
#include <tuple>
#include <random>
#include <ctime>
#include <string>

using namespace std;

vector<vector<int>> dp;

int main()
{
#ifdef _DEBUG
        freopen("input.txt", "r", stdin);
#endif
        ios::sync_with_stdio(0);
        cin.tie(0);
        cout.tie(0);
        srand(time(0));

        int n, x, y;
        cin >> n >> x >> y;
        vector<int> v(n + 1), w(n + 1);
        for (int i = 1; i <= n; i++)
                cin >> v[i];
        for (int i = 1; i <= n; i++)
                cin >> w[i];
        dp.resize(n + 1, vector<int>(x + 1, 1e9));
        for (int i = 0; i <= x; i++)
                dp[0][i] = 0;
        for (int i = 1; i <= n; i++)
        {
                for (int V = 0; V <= x; V++)
                {
                        dp[i][V] = dp[i - 1][V] + w[i];
                        if (V - v[i] >= 0 && dp[i - 1][V - v[i]] < dp[i][V])
                                dp[i][V] = dp[i - 1][V - v[i]];
                }
        }
        int V = -1;
        for (int i = 0; i <= x; i++)
                if (dp[n][i] <= y)
                        V = i;
        if (V == -1)
        {
                cout << -1;
                return 0;
        }
        vector<char> ans;
        for (int i = n; i >= 1; i--)
        {
                if (dp[i - 1][V] + w[i] == dp[i][V])
                        ans.push_back('y');
                else
                        V -= v[i], ans.push_back('x');
        }
        reverse(ans.begin(), ans.end());
        for (char c : ans)
                cout << c;
        return 0;
}
```

Task D ()

```cpp
#define _CRT_SECURE_NO_WARNINGS

#include <set>
#include <unordered_set>
#include <map>
#include <iostream>
#include <vector>
#include <algorithm>
#include <queue>
#include <tuple>
#include <random>
#include <ctime>
#include <string>

using namespace std;

int main()
{
#ifdef _DEBUG
        freopen("input.txt", "r", stdin);
#endif
        ios::sync_with_stdio(0);
        cin.tie(0);
        cout.tie(0);
        srand(time(0));

        int n;
        cin >> n;
        string s;
        cin >> s;
        vector<int> bal;
        for (char c : s)
        {
                if (c == '(' || c == ')')
                {
                        if (!bal.empty() && bal.back() == 0)
                                bal.pop_back();
                        else
                                bal.push_back(0);
                }
                else
                {
                        if (!bal.empty() && bal.back() == 1)
                                bal.pop_back();
                        else
                                bal.push_back(1);
                }
        }
        int ans = 0;
        n = bal.size();
        for (int i = 0; i < n - i - 1; i++)
                ans += bal[i] != bal[n - i - 1];
        cout << ans;
        return 0;
}
```

Task E ()

```
#define _CRT_SECURE_NO_WARNINGS

#include <set>
#include <unordered_set>
#include <map>
#include <iostream>
#include <vector>
#include <algorithm>
#include <queue>
#include <tuple>
#include <random>
#include <ctime>
#include <string>

using namespace std;

int main()
{
#ifdef _DEBUG
        freopen("input.txt", "r", stdin);
#endif
        ios::sync_with_stdio(0);
        cin.tie(0);
        cout.tie(0);
        srand(time(0));

        vector<int> from(10000, -1), to(1000, -1);
        for (int i = 0; i < 10; i++)
        {
                for (int j = i + 1; j < 10; j++)
                {
                        for (int l = j + 1; l < 10; l++)
                        {
                                bool mark = 0;
                                for (int k = 1; k <= 10; k++)
                                {
                                        if (k == i || k == j || k == l)
                                                continue;
                                        int res = 1;
                                        vector<int> v = { i, j, l, k };
                                        sort(v.begin(), v.end());
                                        res *= v[0] * 1000 + v[1] * 100 + v[2] * 10 + v[3];
                                        if (from[res] == -1)
                                        {
                                                mark = 1;
                                                from[res] = k;
                                                to[i * 100 + j * 10 + l] = k;
                                                break;
                                        }
                                }
                                if (!mark)
                                        cout << i << '␣' << j << '␣' << l << '\n';
                        }
                }
        }
        string type;
        cin >> type;
        int t;
        cin >> t;
        int val = 152349;
        for (int i = 0; i < t; i++)
        {
                int n, k;
                cin >> n >> k;
                vector<int> v(k);
                for (int i = 0; i < k; i++)
                        cin >> v[i], v[i]--;
                sort(v.begin(), v.end());
                if (n == 10)
                {
                        if (type == "add")
                                cout << to[v[0] * 100 + v[1] * 10 + v[2]] + 1 << '\n';
```

```
            else
            {
                int o;
                cin >> o;
                v.push_back(o - 1);
                sort(v.begin(), v.end());
                int res = v[0] * 1000 + v[1] * 100 + v[2] * 10 + v[3];
                for (int i = 0; i <= k; i++)
                    if (v[i] != from[res])
                        cout << v[i] + 1 << '␣';
                cout << '\n';
            }
            continue;
        }
        if (type == "add")
            cout << val;
        else
        {
            for (int i = 0; i < k; i++)
                if (v[i] + 1 != val)
                    cout << v[i] + 1 << '␣';
            int o;
            cin >> o;
            if (o != val)
                cout << o;
        }
    }
    return 0;
}
```

Task F ()

```
#define _CRT_SECURE_NO_WARNINGS

#include <set>
#include <unordered_set>
#include <map>
#include <iostream>
#include <vector>
#include <algorithm>
#include <queue>
#include <tuple>
#include <random>
#include <ctime>
#include <string>

using namespace std;

vector<vector<int>> dp;

int main()
{
#ifdef _DEBUG
        freopen("input.txt", "r", stdin);
#endif
        ios::sync_with_stdio(0);
        cin.tie(0);
        cout.tie(0);
        srand(time(0));

        int n;
        cin >> n;
        if (n > 8)
                return -1;
        cout << 4 << '\n';
        cout << "0␣0\n1␣0\n1␣1\n0␣1\n";
        vector<pair<int, int>> v(8);
        v[0] = { -1, -1 };
        v[1] = { -1, 0 };
        v[2] = { -1, 1 };
        v[3] = { 0, 1 };
        v[4] = { 1, 1 };
        v[5] = { 1, 0 };
        v[6] = { 1, -1 };
        v[7] = { 0, -1 };
        for (int i = 0; i < n; i++)
                cout << v[i].first << '␣' << v[i].second << '\n';
        return 0;
}
```