Олимпиада СПбГУ по информатике 2020/21 учебного года

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

Task A ()

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

int main()
{
        int n; cin >> n;
        string h = "023456789";
        if (n <= 9)
                cout << n;
        else
                cout << h[(n - 1) % 9];
        cout << '\n';

        return 0;
}
```

Task B ()

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

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

        int res = 0, tcnt = 0;
        vector<char> cnt('z' - 'a' + 1);
        for (int i = 0; i < n;)
        {
                for (int len = 1; i < n && len <= k; ++i, ++len)
                {
                        if (!cnt[s[i] - 'a'])
                                ++tcnt;

                        cnt[s[i] - 'a'] = 1;

                        if (tcnt > 3) break;
                }
                ++res;
                cnt.assign('z' - 'a' + 1, 0);
                tcnt = 0;
        }

        cout << res << '\n';

        return 0;
}
```

Task C ()

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

int main()
{
	int n, x, y;
	cin >> n >> x >> y; ++x; ++y;
	vector<int> a(n), b(n);
	for (auto& i : a) cin >> i;
	for (auto& i : b) cin >> i;

	vector<int> dp(x);
	vector<vector<char>> p(n, vector<char>(x));

	for (int i = 0; i < x && i < a[0]; ++i)
		dp[i] = b[0], p[0][i] = 1;

	for (int i = 1; i < n; ++i)
	{
		vector<int> ndp(x);
		for (int j = a[i]; j < x; ++j)
			ndp[j] = dp[j - a[i]];

		for (int j = 0; j < x; ++j)
			if (j < a[i] || ndp[j] > dp[j] + b[i])
				ndp[j] = dp[j] + b[i], p[i][j] = 1;

		for (auto& j : ndp)
			j = min(j, y);

		dp = ndp;
	}

	if (dp[x - 1] >= y)
	{
		cout << -1 << '\n';
		return 0;
	}

	string res;
	int i = n - 1, j = x - 1;
	while (i >= 0)
	{
		res += (p[i][j] ? 'y' : 'x');
		j -= (p[i][j] ? 0 : a[i]);
		--i;
	}
	reverse(res.begin(), res.end());
	cout << res << '\n';

	return 0;
}
```

Task D ()

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

int main()
{
	int n; cin >> n; n <<= 1;
	string s; cin >> s;

	stack<char> res;
	for (auto& i : s)
	{
		if (res.size() && abs(res.top() - i) <= 2)
			res.pop();
		else
			res.push(i);
	}

	cout << res.size() / 2 << '\n';

	return 0;
}
```

Task E ()

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

int c1 = 123238, c3 = 23;

int main()
{
        string reg; cin >> reg;
        vector<int> l(1000, -1), r(10000, -1);

        for (int i = 0; i < 10; ++i)
                for (int j = i + 1; j < 10; ++j)
                        for (int k = j + 1; k < 10; ++k)
                        {
                                for (int d = 0; d < 10; ++d)
                                {
                                        if (i == d || j == d || k == d) continue;
                                        vector<int> tek{ i, j, k, d };
                                        sort(tek.begin(), tek.end());

                                        int sum = tek[0] * 1 + tek[1] * 10 + tek[2] * 100 + tek[3]
                                             * 1000;
                                        if (r[sum] != -1) continue;

                                        l[i * 1 + j * 10 + k * 100] = r[sum] = d;
                                        break;
                                }
                        }

        if (reg == "add")
        {
                int t; cin >> t;
                while (t--)
                {
                        int n, k; cin >> n >> k;
                        vector<int> a(k);
                        for (auto& i : a) cin >> i;
                        sort(a.begin(), a.end());

                        if (n == 1000000)
                        {
                                cout << c1 << '\n';
                        }

                        if (n == 10)
                        {
                                cout << l[a[0] * 1 + a[1] * 10 + a[2] * 100 - 1 - 10 - 100] + 1 <<
                                     '\n';
                        }

                        if (n == 100000)
                        {
                                vector<int> cnt(100000 + 1);
                                for (auto& i : a) ++cnt[i];

                                int targ;
                                for (int i = 1; i <= 100000; ++i)
                                {
                                        int sum = 0;
                                        for (int j = i; j < i + c3; ++i)
                                                sum += cnt[j];
                                        if (!sum) continue;

                                        targ = i;
                                        break;
                                }
                                cout << targ << '\n';
                        }
                }
        }

        if (reg == "clear")
```

```
	{
		int t; cin >> t;
		while (t--)
		{
			int n, k; cin >> n >> k;
			vector<int> a(k + 1);
			for (auto& i : a) cin >> i;
			sort(a.begin(), a.end());

			if (n == 1000000)
			{
				for (auto& i : a)
					if (i != c1)
						cout << i << '␣';

				cout << '\n';
			}

			if (n == 10)
			{
				int imposter = r[a[0] * 1 + a[1] * 10 + a[2] * 100 + a[3] * 1000 -
				    1 - 10 - 100 - 1000] + 1;
				for (auto&i: a)
					if (i != imposter)
						cout << i << '␣';
				cout << '\n';
			}

			if (n == 100000)
			{
				vector<int> cnt(100000 + 1);
				for (auto& i : a) ++cnt[i];

				int targ;
				for (int i = 1; i <= 100000; ++i)
				{
					if (cnt[i] != 1) continue;
					int sum = 0;
					for (int j = i + 1; j < i + c3; ++i)
						sum += cnt[j];
					if (sum) continue;

					targ = i;
					break;
				}

				for (auto& i : a)
					if (i != targ)
						cout << i << '␣';
				cout << '\n';
			}
		}

	}

	return 0;
}
```

Task F ()

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

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

        cout << "4\n";
        cout << "0 0\n";
        cout << "0 1\n";
        cout << "1 1\n";
        cout << "1 0\n";

        vector<int> x{-1, -1, -1,  0, 0,  1, 1, 1};
        vector<int> y{-1,  0,  1, -1, 1, -1, 0, 1};

        for (int i = 0; i < n; ++i)
                cout << x[i] << ' ' << y[i] << '\n';

        return 0;
}
```