

# Олимпиада СПбГУ по информатике 2022/23 учебного года

A	B	C	D	E	F	Sum
100	100	100	10	24	65	399

## Task A ()

```
#pragma GCC optimize("Ofast")
#pragma GCC target("avx,avx2,fma")
#include <iostream>
#include <vector>
#include <string>
#include <algorithm>
#include <cmath>

using namespace std;

signed main() {
    ios::sync_with_stdio(0);
    cin.tie(0);
    cout.tie(0);

    vector<int> a(6);
    for (int i = 0; i < 6; i++) {
        cin >> a[i];
    }
    vector<int> v = {1, 2, 3, 4, 5, 6};
    int n = 6;
    while (true) {
        bool good = true;
        for (int i = 0; i < n; i++) {
            int kol = 1;
            for (int j = 0; j < i; j++) {
                if (v[j] < v[i]) {
                    kol++;
                }
            }
            if (kol != a[i]) {
                good = false;
                break;
            }
        }
        if (good) {
            for (int el : v) {
                cout << el << ' ';
            }
            cout << endl;
            return 0;
        }
        next_permutation(v.begin(), v.end());
    }
}
```

## Task B ()

```
#pragma GCC optimize("Ofast")
#pragma GCC target("avx,avx2,fma")
#include <iostream>
#include <vector>
#include <string>
#include <algorithm>
#include <cmath>

using namespace std;
#define int long long
#define ll long long

signed main() {
    ios::sync_with_stdio(0);
    cin.tie(0);
    cout.tie(0);

    string s;
    cin >> s;
    int n;
    cin >> n;
    vector<ll> v(n);
    for (int i = 0; i < n; i++) {
        cin >> v[i];
    }
    if (s == "first") {
        ll sum = 0;
        for (auto el : v) {
            sum += el;
        }
        sum *= (ll) 1e9;
        cout << sum << endl;
        return 0;
    } else {
        ll x = 0;
        ll now = v[0];
        vector<int> a;
        while (now > 0) {
            a.push_back(now % 10);
            now = (now - now % 10) / 10;
        }
        vector<int> num;
        for (int i = 9; i < a.size(); i++) {
            num.push_back(a[i]);
        }
        //reverse(num.begin(), num.end());
        ll mult = 1;
        for (auto el : num) {
            x += mult * (ll) el;
            mult *= 10;
        }
        ll ans = 0;
        for (auto el : v) {
            ans += el - x * (ll) 1e9;
        }
        cout << ans + x;
        cout << endl;
        return 0;
    }
}

/*

second
5
15000000006 15000000007 15000000008 15000000009 15000000010

15000000000
*/
```

## Task C ()

```
#pragma GCC optimize("Ofast")
#pragma GCC target("avx,avx2,fma")
#include <iostream>
#include <vector>
#include <string>
#include <algorithm>
#include <cmath>

using namespace std;

signed main() {
    ios::sync_with_stdio(0);
    cin.tie(0);
    cout.tie(0);

    struct node {
        int a;
        int b;
    };
    vector<node> v(3);
    vector<pair<int, int>> xs(3);
    int n = 3;
    for (int i = 0; i < n; i++) {
        int a, b;
        cin >> a >> b;
        if (a > b) {
            swap(a, b);
        }
        v[i] = {a, b};
        xs[i] = {a * b, i};
    }
    sort(xs.rbegin(), xs.rend());
    auto v2 = v;
    v[0] = v2[xs[0].second];
    v[1] = v2[xs[1].second];
    v[2] = v2[xs[2].second];
    vector<pair<int, int>> ans;
    /*for (int i1 = 0; i1 < 27; i1++) {
        for (int j1 = 0; j1 < 27; j1++) {
            for (int i2 = i1; i2 < 27; i2++) {
                for (int j2 = ; j2 < 27; j2++) {

                }
            }
        }
    }*/
    ans.push_back({v[0].a, v[0].b});
    ans.push_back({v[1].a, v[1].b});
    ans.push_back({v[2].a, v[2].b});

    for (int i = 0; i < n; i++) {
        for (int j = i + 1; j < n; j++) {
            if (v[i].a == v[j].a) {
                int a = v[i].a;
                int b = v[i].b - v[j].b;
                ans.push_back({a, b});
            }
            if (v[i].a == v[j].b) {
                int a = v[i].a;
                int b = v[i].b - v[j].a;
                ans.push_back({a, b});
            }
            if (v[i].b == v[j].b) {
                int a = v[i].b;
                int b = v[i].a - v[j].a;
                ans.push_back({a, b});
            }
            // v[i].b > v[j].a
        }
    }

    for (int qq = 0; qq < 1000; qq++) {
```

```

    if (v[0].a == v[1].a + v[2].a && v[0].b >= v[1].b && v[0].b >= v[2].b && v[1].b + v[2].b
        == v[0].b) {
        ans.push_back({v[1].a, v[2].b});
        ans.push_back({v[2].a, v[1].b});
    }
    if (v[0].a == v[1].a + v[2].a && v[1].b == v[2].b) {
        ans.push_back({v[0].a, v[0].b - v[1].b});
    }

    if (v[0].a < v[1].b + v[2].a && v[2].b + v[1].a <= v[0].b) {
        ans.push_back({v[0].a - v[1].b, v[1].a});
    }
    if (v[0].a < v[2].b + v[1].a && v[1].b + v[2].a <= v[0].b) {
        ans.push_back({v[0].a - v[2].b, v[2].a});
    }
    for (int i = 0; i < n; i++) {
        swap(v[i].a, v[i].b);
    }
    if (v[0].a == v[1].a + v[2].a && v[0].b >= v[1].b && v[0].b >= v[2].b && v[1].b + v[2].b
        == v[0].b) {
        ans.push_back({v[1].a, v[2].b});
        ans.push_back({v[2].a, v[1].b});
    }
    if (v[0].a == v[1].a + v[2].a && v[1].b == v[2].b) {
        ans.push_back({v[0].a, v[0].b - v[1].b});
    }

    if (v[0].a < v[1].b + v[2].a && v[2].b + v[1].a <= v[0].b) {
        ans.push_back({v[0].a - v[1].b, v[1].a});
    }
    if (v[0].a < v[2].b + v[1].a && v[1].b + v[2].a <= v[0].b) {
        ans.push_back({v[0].a - v[2].b, v[2].a});
    }
    for (int i = 0; i < n; i++) {
        swap(v[i].a, v[i].b);
    }
    for (int i = 0; i < ans.size(); i++) {
        if (ans[i].first > ans[i].second) {
            swap(ans[i].first, ans[i].second);
        }
    }
    for (int qq = 0; qq < 1000; qq++) {
        int k = rand() % 3;
        swap(v[k].a, v[k].b);
    }
}
sort(ans.begin(), ans.end());
ans.erase(unique(ans.begin(), ans.end()), ans.end());
for (auto el : ans) {
    if (min(el.first, el.second) <= 0) {
        continue;
    }
    cout << el.first << ' ' << el.second << '\n';
}
}

```

## Task D ()

```
#pragma GCC optimize("Ofast")
#pragma GCC target("avx,avx2,fma")
#include <iostream>
#include <vector>
#include <string>
#include <algorithm>
#include <cmath>
#include <deque>

using namespace std;

vector<int> vst;
vector<int> v;
vector<int> u;

void ask(int i, int x) {
    cout << i + 1 << ' ' << x << endl;
    if (x == 0) {
        v[i] = vst[i];
        u[i] = 1;
    } else {
        v[i] -= x;
    }
    int i1, x1;
    cin >> i1 >> x1;
    if (i1 == -1 && x1 == -1) {
        exit(0);
    }
    i1--;
    if (x1 == 0) {
        v[i1] = vst[i1];
        u[i1] = 1;
    } else {
        v[i1] -= x1;
    }
}

signed main() {
    ios::sync_with_stdio(0);
    cin.tie(0);
    cout.tie(0);

    int n;
    cin >> n;
    v.resize(n);
    u.resize(n, 0);
    for (int i = 0; i < n; i++) {
        cin >> v[i];
    }
    vst = v;

    vector<vector<vector<int>>> g(n, vector<vector<int>> (103));
    vector<vector<vector<int>>> rg(n, vector<vector<int>> (103));
    vector<vector<int>> mex(n, vector<int> (103, 0));
    for (int i = 0; i < n; i++) {
        for (int j = 0; j <= v[i]; j++) {
            g[i][j].push_back(v[i] + 51);
            rg[i][v[i] + 51].push_back(j);
            for (int to = 0; to < j; to++) {
                g[i][j].push_back(to);
                rg[i][to].push_back(j);
            }
        }
        for (int j = 0; j <= v[i]; j++) {
            for (int to = 0; to < j; to++) {
                g[i][j + 51].push_back(to + 51);
                rg[i][to + 51].push_back(j + 51);
            }
        }
    }

    deque<int> d;
    vector<int> kol(103, 0);
```



## Task E ()

```
#pragma GCC optimize("Ofast")
#pragma GCC target("avx,avx2,fma")
#include <iostream>
#include <vector>
#include <string>
#include <algorithm>
#include <cmath>

using namespace std;

vector<int> conv(int x) {
    int xs = x;
    vector<int> v;
    for (int i = 2; i * i <= xs; i++) {
        while (x % i == 0) {
            v.push_back(i);
            x /= i;
        }
    }
    if (x > 1) {
        v.push_back(x);
    }
    return v;
}

int pr[(int) 1e5];
vector<int> prime;

vector<int> get(int x) {
    int ans = 0;
    vector<int> maxx = {};
    for (int i = 0; i < (int) 1e2; i++) {
        vector<int> v;
        for (int j = 0; j < 10; j++) {
            if (x % prime[i + j] > 10) {
                continue;
            }
            if ((int) v.size() > 0 && v[(int) v.size() - 1] > x % prime[i + j]) {
                break;
            }
            v.push_back(x % prime[i + j]);
        }
        if (ans < (int) v.size()) {
            maxx = v;
            ans = (int) v.size();
        }
    }
    return maxx;
}

signed main() {
    ios::sync_with_stdio(0);
    cin.tie(0);
    cout.tie(0);

    memset(pr, 0, sizeof(pr));
    for (int i = 2; i < (int) 1e5; i++) {
        if (pr[i] == 0) {
            pr[i] = i;
            prime.push_back(i);
        }
        for (int j = 0; j < prime.size() && prime[j] * i < (int) 1e5; j++) {
            pr[prime[j] * i] = prime[j];
        }
    }
    int t;
    cin >> t;
    string s;
    cin >> s;
    if (s == "transmit") {
        for (int q = 0; q < t; q++) {
```





1000000000  
0000000000  
0000000000  
0000000000  
0000000000  
0000000000  
0000000000  
0000000000  
0000000000

0000000000  
1111100000  
1111100000  
1111100000  
1111100000  
1111100000  
1111100000  
1111100000  
1111100000  
1111100000

0000000000  
1100000000  
1100000000  
1100000000  
1100000000  
1100000000  
1100000000  
1100000000  
1100000000  
1100000000

\*/

## Task F ()

```
#pragma GCC optimize("Ofast")
#pragma GCC target("avx,avx2,fma")
#include <iostream>
#include <vector>
#include <string>
#include <algorithm>
#include <cmath>

using namespace std;
#define ll long long
#define int long long

signed main() {
    ios::sync_with_stdio(0);
    cin.tie(0);
    cout.tie(0);

    for (int qq = 0; qq < 1; qq++) {
        string a = "", b = "";
        cin >> a >> b;
        /*int as = rand() % 100 + 1;
        int bs = rand() % 100 + 1;
        for (int i = 0; i < as; i++) {
            a += to_string(rand() % 10);
        }
        for (int i = 0; i < bs; i++) {
            b += to_string(rand() % 10);
        }*/
        vector<pair<int, ll>> v1;
        vector<pair<int, ll>> v2;
        ll sz1 = 0;
        ll sz2 = 0;
        for (int i = 0; i < a.size(); i++) {
            if (a[i] == '(') {
                string kol = "";
                int last = i + 3;
                for (int j = i + 3; a[j] != ')'; j++) {
                    kol += a[j];
                    last = j;
                }
                v1.push_back({(int) (a[i + 1] - '0'), stoll(kol)});
                sz1 += stoll(kol);
                i = last + 1;
            } else {
                sz1++;
                v1.push_back({(int) (a[i] - '0'), 1});
            }
        }
        for (int i = 0; i < b.size(); i++) {
            if (b[i] == '(') {
                string kol = "";
                int last = i + 3;
                for (int j = i + 3; b[j] != ')'; j++) {
                    kol += b[j];
                    last = j;
                }
                v2.push_back({(int) (b[i + 1] - '0'), stoll(kol)});
                sz2 += stoll(kol);
                i = last + 1;
            } else {
                v2.push_back({(int) (b[i] - '0'), 1});
                sz2++;
            }
        }
        if (sz1 < sz2) {
            reverse(v1.begin(), v1.end());
            v1.push_back({0, sz2 - sz1});
            reverse(v1.begin(), v1.end());
        } else if (sz2 < sz1) {
            reverse(v2.begin(), v2.end());
            v2.push_back({0, sz1 - sz2});
            reverse(v2.begin(), v2.end());
        }
    }
}
```

```

}

int l = v1.size() - 1;
int k1 = v1[l].second;
int r = v2.size() - 1;
int k2 = v2[r].second;
int add = 0;
vector<pair<int, int>> res;
while (l >= 0 && r >= 0) {
    if (k1 == 0) {
        l--;
        if (l >= 0) {
            k1 = v1[l].second;
        }
        continue;
    }
    if (k2 == 0) {
        r--;
        if (r >= 0) {
            k2 = v2[r].second;
        }
        continue;
    }

    k1--;
    k2--;
    res.push_back({(v1[l].first + v2[r].first + add) % 10, 1});
    if (v1[l].first + v2[r].first + add >= 10) {
        add = 1;
    } else {
        add = 0;
    }
    if (k1 == 0) {
        l--;
        if (l >= 0) {
            k1 = v1[l].second;
        }
        continue;
    }
    if (k2 == 0) {
        r--;
        if (r >= 0) {
            k2 = v2[r].second;
        }
        continue;
    }

    int minn = min(k1, k2);
    res.push_back({(v1[l].first + v2[r].first + add) % 10, minn});
    if (v1[l].first + v2[r].first + add >= 10) {
        add = 1;
    } else {
        add = 0;
    }
    k1 -= minn;
    k2 -= minn;

    if (k1 == 0) {
        l--;
        if (l >= 0) {
            k1 = v1[l].second;
        }
        continue;
    }
    if (k2 == 0) {
        r--;
        if (r >= 0) {
            k2 = v2[r].second;
        }
        continue;
    }
}
}
/*while (l >= 0) {
    if (add) {

```

```

        k1--;
        res.push_back({(v1[l].first + 1) % 10, 1});
        if (v1[l].first + 1 >= 10) {
            add = 1;
        } else {
            add = 0;
        }
    }

    if (k1 == 0) {
        l--;
        if (l >= 0) {
            k1 = v1[l].second;
        }
        continue;
    }

    int minn = k1;
    res.push_back({(v1[l].first + add) % 10, minn});
    if (v1[l].first + add >= 10) {
        add = 1;
    } else {
        add = 0;
    }
    k1 -= minn;

    if (k1 == 0) {
        l--;
        if (l >= 0) {
            k1 = v1[l].second;
        }
        continue;
    }
}

while (r >= 0) {
    if (add) {
        k2--;
        res.push_back({(v2[r].first + 1) % 10, 1});
        if (v2[r].first + 1 >= 10) {
            add = 1;
        } else {
            add = 0;
        }
    }

    if (k2 == 0) {
        r--;
        if (r >= 0) {
            k2 = v2[r].second;
        }
        continue;
    }

    int minn = k2;
    res.push_back({(v2[r].first + add) % 10, minn});
    if (v2[r].first + add >= 10) {
        add = 1;
    } else {
        add = 0;
    }
    k2 -= minn;

    if (k2 == 0) {
        r--;
        if (r >= 0) {
            k2 = v2[r].second;
        }
        continue;
    }
}
}*/
if (add) {
    res.push_back({1, 1});
}

```

```

reverse(res.begin(), res.end());
for (auto el: res) {
    if (el.second == 0) {
        continue;
    }
    cout << '(' << el.first << '|' << el.second << ')';
}
}
}
/*
18
97
*/

```